FUJITSU

SIMPLIA/MF-STEPCOUNTER

オンラインマニュアル

（プログラムステップ計測ツール）

SIMPLIA/MF-STEPCOUNTER V50L42
オンラインマニュアル

第 1.0 版

はじめに

SIMPLIA/MF-STEPCOUNTERは、C、COBOL、Java、IDL、HTML、JSP、CSS、JavaScript、ASP、C#、Visual Basic、VBScript、JScriptのソースファイル、インクルードファイル、COBOL登録集をもとに、個々のプログラムステップ情報の計測と、各開発資源の相互関係を解析し、ドキュメントを作成するアプリケーションです。

新機能

- COBOLの拡張子"*.scob"および"*.pco"に対応しました。
- COBOLの副プログラムの計測制限を解除しました。 1つのソースファイル中に複数のプログラムが書かれている場合にも計測できます。
- C/C++のフォルダ指定での計測で、"*.h"および"*.rc"を含めて計測するようになりました。
- リストビュー形式の計測結果画面の列幅表示を改善しました。
- リストビュー形式の計測結果画面で、計測結果のクリップボードへのコピーができるようになりました。

ヘルプを読むために

HTML3.2をサポートするWWWブラウザ（インターネットエクスプローラ V3.02以降、Netscape NavigatorV4.03以降）をお使いください。

登録商標について

本オンラインマニュアルで使われている登録商標及び商標は、以下のとおりです。

- Microsoft,Windows,MS-DOS,MS,Visual C#,Visual Basicは、米国Microsoft Corporationの米国およびその他の国における登録商標です。
- ORACLEは、米国Oracle Corporationの登録商標です。
- INFORMIXは、米国Informix Software,Inc.の登録商標です。
- JavaおよびすべてのJava関連の商標およびロゴは、米国およびその他の国における米国Sun Microsystems,Inc.の商標または登録商標です。

略記について

本オンラインマニュアルでは、各製品を次のように略記しています。

「Microsoft(R) Windows(R) 2000 Professional operating system」	→	「Windows 2000 Pro」
「Microsoft(R) Windows(R) 2000 Server operating system」	→	「Windows 2000 Server」
「Microsoft(R) Windows(R) 2000 Advanced Server operating system」	→	「Windows 2000 Advanced Server」
「Microsoft(R) Windows Server(R) 2003, Standard Edition」 「Microsoft(R) Windows Server(R) 2003 R2, Standard Edition」 「Microsoft(R) Windows Server(R) 2003, Enterprise Edition」 「Microsoft(R) Windows Server(R) 2003 R2, Enterprise Edition」	→	「Windows Server 2003」
「Microsoft(R) Windows Server(R) 2008 Standard」 「Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM)」 「Microsoft(R) Windows Server(R) 2008 Enterprise」 「Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM)」	→	「Windows Server 2008」
「Microsoft(R) Windows(R) XP Home Edition operating system」	→	「Windows XP」
「Microsoft(R) Windows(R) XP Professional operating system」	→	「Windows XP」
「Microsoft(R) Windows Vista(R) Enterprise」	→	「Windows Vista」
「Microsoft(R) Visual C#(R)」	→	「C#」
「Microsoft(R) Visual Basic(R) Version 6.0」	→	「Visual Basic 6.0」 または、「VB6.0」
「Java(TM)」	→	「Java」

導入の手引き

STEPCOUNTERのインストール

STEPCOUNTERのインストールについては、製品に付属するソフトウェア説明書を参照してください。なお、ソフトウェア説明書は製品ディスクにテキストファイル「README.TXT」として格納されています。

STEPCOUNTERをはじめてお使いになる方は、サンプルの使い方を参照ください。一通りの操作について説明しています。

メトリクス計測機能の使用方法について

メトリクス計測機能の使用方法については、以下のマニュアルを参照してください。

SIMPLIA/MF-STEPCOUNTER Javaソフトウェアメトリクス計測機能 V50L42 オンラインマニュアル

メニューコマンド

その他

[ファイル]メニューのコマンド

ファイル(F)	計測(K)	表示(V)	オプション(O)	ウィンドウ(W)

メニューコマンド	機能
新規計測	新規に計測を行い、計測結果を画面に表示します。
読み込み	既存の計測結果ファイルを読み込み、計測結果を画面に表示します。
対象資産の編集	指定した計測対象の一覧に対して追加、削除等の編集を行います。
上書き保存	計測結果をファイルへ上書き保存します。
名前を付けて保存	計測結果をファイルへ名前を指定して保存します。
選択範囲のみCSV形式ファイルの作成	計測結果の表示形式がリストビュー形式で表示されている場合に、選択されている対象ファイル行のみCSV形式のファイルへ保存します。（リストビュー形式時のみ可能。表示形式の変更については表示オプションの設定を参照してください。）
CSV形式ファイルの作成	計測結果の内容をCSV形式のファイルへ保存します。 帳票形式時のCSV形式ファイル リストビュー形式時のCSV形式ファイル
印刷	印刷に関する情報を設定し、プリンタへ計測結果を出力します。（計測結果の表示形式が帳票形式時のみ可能。表示形式の変更については表示オプションの設定を参照してください。）
ファイル履歴	過去に扱った計測結果ファイルを新しいものから最大５件表示します。選択された計測結果ファイルを読み込み、計測結果を画面に表示します。

 関連項目

[計測]メニューのコマンド

ファイル(F)	計測(K)	表示(V)	オプション(O)	ウィンドウ(W)

メニューコマンド	機能
全計測	指定されている全計測対象ファイルに対して再計測を行い、計測結果を画面に表示します。オプションの変更を計測結果に反映したい場合は、このコマンドを実行する必要があります。
変更部分のみ計測	「対象資産の編集」処理で追加したファイルに対してのみ計測を行い、計測結果を表示します。 「対象資産の編集」処理で削除した場合は本コマンドは使用できません。全計測を行なってください。

関連項目

[表示]メニューのコマンド

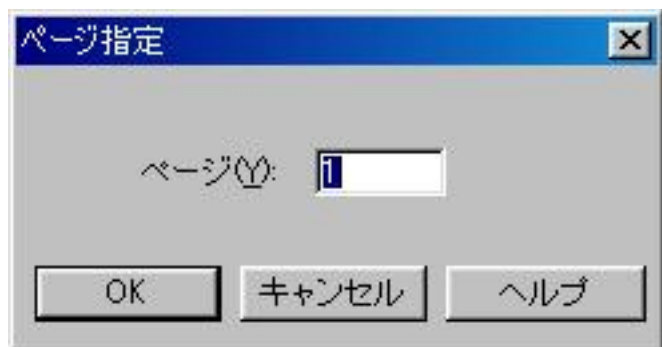
ファイル(F)	計測(K)	表示(V)	オプション(O)	ウィンドウ(W)

メニューコマンド	機能
ツールバー	メイン／表示制御／表示フォルダの各ツールバーの表示／非表示を切り替えます。
ステータスバー	ステータスバーの表示／非表示を切り替えます。
ファイル情報	ファイル情報の種類、サイズ、更新日付の表示／非表示を切り替えます。（※１）
SQLステップ数	SQL情報のSQL、ETCの表示／非表示を切り替えます。（※１）
SQLの比率	SQL情報のSQL率の表示／非表示を切り替えます。（※１）
注釈ステップ数	注釈、JavaDoc、ETCの表示／非表示を切り替えます。（※１）
注釈の比率	注釈率、Javadoc率の表示／非表示を切り替えます。（※１）
組込み形態別	プログラム情報では、組込み形態別項目、組込みメンバ情報では、組込み情報の表示／非表示を切り替えます。（※１）
フォルダ名	フォルダ名の表示／非表示を切り替えます。（※１）
初期表示に戻す	各列の表示／非表示を表示オプションで指定した状態に戻します。（※１）
前ページ	現在アクティブなドキュメントウィンドウの表示ページを一つ前に戻します。（※２）
次ページ	現在アクティブなドキュメントウィンドウの表示ページを一つ後に進めます。（※２）
先頭ページ	現在アクティブなドキュメントウィンドウで先頭ページを表示します。（※２）
最終ページ	現在アクティブなドキュメントウィンドウで最終ページを表示します。（※２）
ページ指定	現在アクティブなドキュメントウィンドウで指定ページを表示します。（※２）

※１　計測結果の表示形式がリストビュー形式時のみ可能です。
※２　計測結果の表示形式が帳票形式時のみ可能です。

 関連項目

ページ指定 ダイアログボックス

帳票形式の計測結果表示画面で、表示したいページを指定します。

項目	説明
[ページ]	表示したいページを[ページ]エディットボックスに設定します。

[オプション]メニューのコマンド

ファイル(F)	計測(K)	表示(V)	オプション(O)	ウィンドウ(W)

メニューコマンド	機能
計測オプション	計測の種別及び情報について設定します。
表示オプション	画面の表示形式及びリストビュー形式画面に関する情報を指定します。
サーチパス	ソースプログラムで展開されている組込みファイルが存在するフォルダのパスを指定します。
CSVファイル	CSV形式ファイルの出力に関する情報を指定します。
帳票オプション	帳票形式画面及び印刷帳票に関する情報を指定します。
Java自動識別オプション	Java自動生成ソース計測時の手書き部分／自動生成部分の計測に関する情報を指定します。
VB6.0自動識別オプション	Visual Basic 6.0ソース計測時の手書き部分／自動生成部分の計測に関する情報を指定します。

 関連項目

[ウィンドウ]メニューのコマンド

COBOLやC/C++など計測結果の表示で、プログラムステップ情報ウィンドウ以外に組込みメンバステップ情報ウィンドウが表示されるような、複数のウィンドウが表示される場合に使用します。

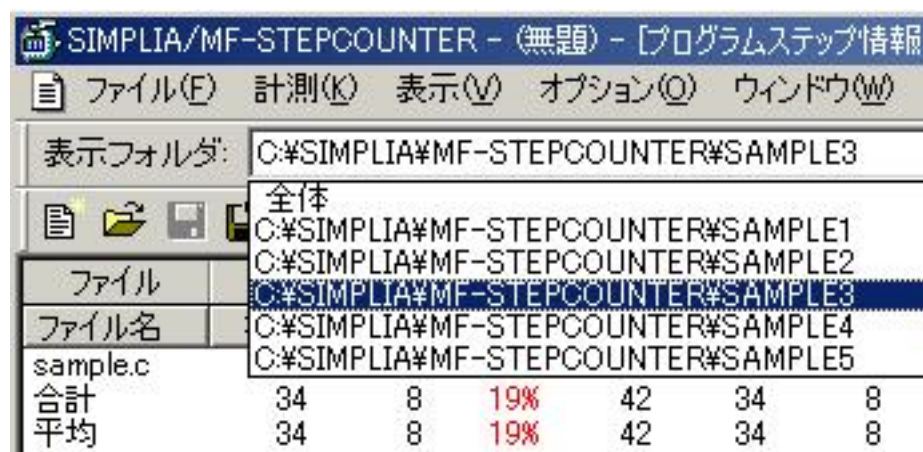
メニューコマンド	機能
重ねて表示	すべてのウィンドウのタイトルバーが見えるように、開いているウィンドウを少しずつずらして並べます。
縦に並べて表示	すべてのウィンドウが見えるように、開いているウィンドウを上下に敷きつめて並べます。
横に並べて表示	すべてのウィンドウが見えるように、開いているウィンドウを左右に敷きつめて並べます。

その他 現在表示されているドキュメントウィンドウのタイトルバー文字列がメニューサブ項目として追加されます。追加されたサブメニュー項目を選択することにより、希望のドキュメントウィンドウをアクティブ化できます。

 関連項目

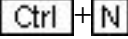
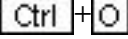
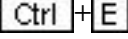
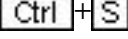
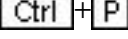
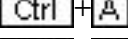
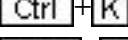
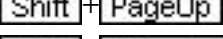
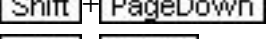
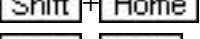
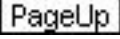
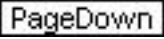
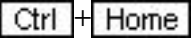
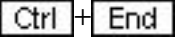
ツールバー

・表示フォルダツールバー

　計測対象資産をサブフォルダを含めた計測を行った場合、「表示フォルダツールバー」のプルダウンメニューには各サブフォルダ名と、全ての計測対象ファイルを意味する「全体」がリストされます。
　このうちの１つを選択することにより、各フォルダ毎の計測結果を表示したり、全ての計測結果を表示したりと切り替えることが出来ます。
　（リストビュー形式時のみ可能）

・メインツールバー

アイコン	対応するメニューコマンド
	新規計測
	読み込み
	上書き保存
	名前を付けて保存
	選択範囲のみCSV形式ファイルの作成
	CSV形式ファイルの作成
	印刷
	全計測
	変更部分のみ計測
	計測オプション
	MF-STEPCOUNTER ヘルプ

・表示制御ツールバー

アイコン	対応するメニューコマンド
	先頭ページ
	前ページ
	次ページ
	最終ページ
	ファイル情報
	SQLステップ数
	SQLの比率
	注釈ステップ数
	注釈の比率
	組込み形態別
	フォルダ名

関連項目

ショートカットキー

メニューコマンドへのショートカットキー

キー操作	対応するメニューコマンド
Ctrl+N	新規計測
Ctrl+O	読み込み
Ctrl+E	対象資産の編集
Ctrl+S	上書き保存
Ctrl+P	印刷
Ctrl+A	全計測
Ctrl+K	変更部分のみ計測
Shift+PageUp	前ページ
Shift+PageDown	次ページ
Shift+Home	先頭ページ
Shift+End	最終ページ

その他のショートカットキー

キー操作	機能
PageUp	現在表示しているページ内でスクロールアップします。
PageDown	現在表示しているページ内でスクロールダウンします。
Ctrl+Home	現在表示しているページの先頭へ移動します。
Ctrl+End	現在表示しているページの最後へ移動します。

関連項目

起動オプション

起動オプションと戻り値について説明します。起動オプションを指定したショートカットやバッチファイルを作成することにより、定型的な計測を定期的に繰り返すことが容易になります。

起動オプション

CMFSTP32.EXE　計測結果ファイル(*.STP)　[-CSV CSVファイル名／出力先フォルダ名]

CMFSTP32.EXE　計測対象ファイル／フォルダ -LANG 言語種別　[-CSV CSVファイル名／出力先フォルダ名]

言語種別には以下のものを指定します。

言語種別	計測基準
COBOL	COBOL計測基準
C	C/C++計測基準
JAVA	Java計測基準
IDL	IDL計測基準
JAVAAUTO	Java自動生成ソース計測基準
HTML	HTML/JSP/JS/CSS計測基準
DOTNET	.NET（ASP/C#/VB）/HTML/JS/CSS計測基準
VB	Visual Basic 6.0計測基準

また、"-CSV"パラメタを指定した場合には、計測終了後に本ツールは自動的に終了します。

以下の例では、インストール先を"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER"にした場合で説明しています。

CMFSTP32.EXEのみ指定した場合

本ツールの起動のみを行います。その後の操作については新規計測の方法または計測結果ファイルの読み込み方法を参照下さい。

使用例

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe"

SIMPLIA/MF-STEPCOUNTERを起動します。

引数に計測結果ファイルを指定した場合

指定された計測結果ファイル(*.STP)を読み込み後再計測し、計測結果を画面またはCSVファイルに出力します。

パラメタの説明

CSV　CSVファイル名(*.CSV)
=>　計測結果をCSVファイルへ出力します。
CSVファイルの出力先に同名のファイルが存在する場合、上書きされます。
計測終了後に、CSVファイルを出力して終了します。

CSV　出力先フォルダ名
=>　計測結果を出力先フォルダへ出力します。出力されるCSVファイル名は"計測年月日時分秒.CSV"になります。
CSVファイルの出力先に同名のファイルが存在する場合、上書きされます。
計測終了後に、CSVファイルを出力して終了します。

使用例

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work¥test.stp

計測結果ファイルtest.stpを再計測し、計測結果を画面出力します。

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work¥test.stp -csv d:¥work¥csv¥test.csv

計測結果ファイルtest.stpを再計測し、計測結果をtest.csvに出力して終了します。

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work¥test.stp -csv d:¥work¥csv

計測結果ファイルtest.stpを再計測し、計測結果を計測年月日時分秒.csvに出力して終了します。

引数に計測対象ファイルを指定した場合

計測対象ファイルを、指定した言語種別の計測基準により、新規計測し、計測結果を出力します。
ただし、計測対象のファイルは単一のファイルしか指定できません。また、ワイルドカード(*や?を使ったファイル指定）による指定もできません。複数のファイルを計測したい場合は、上記の計測結果ファイルを使用するか、下記のフォルダ指定を使用してください。

パラメタの説明

LANG 言語種別
=> 言語種別の計測基準により、新規計測します。
指定されていない場合、パラメタエラーとなります。

CSV CSVファイル名(*.CSV)
=> 計測結果をCSVファイルへ出力して終了します。
CSVファイルの出力先に同名のファイルが存在する場合、上書きされます。
計測終了後に、CSVファイルを出力して終了します。

CSV 出力先フォルダ名
=> 計測結果を出力先フォルダへ出力して終了します。出力されるCSVファイル名は"計測年月日時分秒.CSV"になります。
CSVファイルの出力先に同名のファイルが存在する場合、上書きされます。
計測終了後に、CSVファイルを出力して終了します。

使用例

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work¥test.c -lang c

計測対象ファイルtest.cを言語種別C/C++の計測基準により、新規計測し、計測結果を画面出力します。

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work¥test.c -lang c -csv d:¥work¥csv¥test.csv

計測対象ファイルtest.cを言語種別C/C++の計測基準により、新規計測し、計測結果をtest.csvに出力して終了します。

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work¥test.c -lang c -csv d:¥work¥csv

計測対象ファイルtest.cを言語種別C/C++の計測基準により、新規計測し、計測結果を計測年月日時分秒.csvに出力して終了します。

引数に計測対象フォルダを指定した場合

計測対象フォルダ内のソースファイルを、指定した言語種別の計測基準により、新規計測し、計測結果を出力します。
計測対象となるソースファイルは、指定した言語種別が対象としている拡張子のファイルとなります。

パラメタの説明

LANG　言語種別
=>　言語種別の計測基準により、新規計測します。
指定されていない場合、パラメタエラーとなります。

CSV　CSVファイル名(*.CSV)
=>　計測結果をCSVファイルへ出力します。
CSVファイルの出力先に同名のファイルが存在する場合、上書きされます。
計測終了後に、CSVファイルを出力して終了します。

CSV　出力先フォルダ名
=>　計測結果を出力先フォルダへ出力します。出力されるCSVファイル名は"計測年月日時分秒.CSV"になります。
CSVファイルの出力先に同名のファイルが存在する場合、上書きされます。
計測終了後に、CSVファイルを出力して終了します。

使用例

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work -lang c

計測対象フォルダworkに存在する計測対象ファイルを、言語種別C/C++の計測基準により、新規計測し、計測結果を画面出力します。

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work -lang c -csv d:¥work¥csv¥test.csv

計測対象フォルダworkに存在する計測対象ファイルを、言語種別C/C++の計測基準により、新規計測し、計測結果をtest.csvに出力して終了します。

c:¥>"c:¥Program Files¥SIMPLIA¥MF-STEPCOUNTER¥CMFSTP32.exe" d:¥work -lang c -csv d:¥work¥csv

計測対象フォルダworkに存在する計測対象ファイルを、言語種別C/C++の計測基準により、新規計測し、計測結果を計測年月日時分秒.csvに出力して終了します。

起動オプションの指定が誤っている場合は、エラーファイルをテキストエディタ等で表示します。

戻り値

計測結果出力後の"CMFSTP32.EXE"の戻り値は以下の通りです。

０：正常終了
１：正常終了（計測エラーあり）
９：コマンドラインエラー

注意　計測結果ファイル、計測対象ファイル／フォルダ、およびCSVファイル／フォルダはフルパスで指定してください。ファイル名のみや相対パスは指定しないで下さい。

注意　計測結果ファイルは、１度メイン画面より計測を行って保存した、存在するファイルを指定してください。存在しない場合はエラーになります。

注意　ファイル名やフォルダ名に空白を含む場合は、パスを""で括ってください。
　例）cmftstp32.exe "d:¥work folder¥test.stp"

注意　コマンドプロンプトから実行する場合は、環境変数のPATHに本製品のインストールフォルダを指定していると、exeファイル名による起動が簡単にできます。

新規計測の方法

新規に計測する対象資産をファイル単位または、対象ソースファイルが存在するフォルダ単位で指定し、計測を行います。

新規計測するには

0. C/C++言語およびCOBOL言語資産の計測で、includeやcopy句などの組込みファイルのステップ数を含めた計測を行う場合には、最初にサーチパスオプションの設定により、includeファイルやコピー句ファイルなどのパスを指定してください。また計測オプションの設定で「組込みメンバを計測する」を選択してください。
　ただし、この場合はプログラムファイルと組込みファイルの関係を解析して計測に非常に時間がかかるようになりますので、計測対象の本数を絞るなどしてください。（※）

1. [ファイル]メニューの[新規計測]コマンドを選びます。
2. 新規計測 1/2 ダイアログボックスで[言語種別]、[対象資産の指定方法]及び[文字コードの指定]を選択し、[次へ]ボタンを押下します。
3. 新規計測 2/2 ダイアログボックスで計測対象となるソースファイルまたは、ソースファイルが存在するフォルダを指定します。
4. 全ての計測対象の指定が済んだら、[完了]ボタンを押下すると計測処理が開始され、計測結果を画面に表示します。

※ プログラムファイルと組込みファイルの関係を解析せずに短時間で組込みファイルのステップ数を計測するには、上記の手順0を行わず、手順2の[対象資産の指定方法]で[ファイル]を選択、手順3のソースファイル選択時に[ファイルの種類]を[全てのファイル(*.*)]に設定して組込みファイルを指定し、計測します。

注意　選択した言語種別に応じて、ファイルの拡張子を元に以下の言語種類で計測を行います。種類欄はプログラムステップ情報（リストビュー形式）画面の「種類」欄に表示される名称です。

言語種別	種類	拡張子
COBOL	COBOL	(*.CBL)，(*.COB)，(*.COBOL)，(*.SCOB)，(*.PCO)
C/C++	C/C++	(*.C)，(*.CPP)，(*.H)，(*.RC) （*.Hは計測オプションの設定に依存します。）
Java Java自動生成ソース	Javaクラス Javaインタフェース	(*.JAVA)
	SQLJ	(*.SQLJ)
IDL	IDL	(*.IDL)
HTML/JSP/JS/CSS (注)	HTML	(*.HTML)，(*.HTM)
	JSP	(*.JSP)，(*.JSPF)，(*.TAG)，(*.TAGX)，(*.TAGF)
	VB.NET	(*.VB)
	VBScript	(*.VBS)
	JavaScript	(*.JS)
	StyleSheet	(*.CSS)
.NET（ASP/C#/VB）/HTML/JS/CSS	ASP.NET	(*.ASP),(*.ASPX)
	C#	(*.CS)
	VB.NET	(*.VB)，(*.VBS)

(注)	HTML	(*.HTML)，(*.HTM)
	JavaScript	(*.JS)
	CSS	(*.CSS)
Visual Basic 6.0	VB6.0	(*.FRM)， (*.BAS)， (*.CLS)， (*.CTL)， (*.PAG)， (*.DSR)

注） "JS" は JavaScript と JScript の略です。
"CSS" は Cascading Style Sheets の略です。

ダイアログボックスの操作方法

- 新規計測 １／２ ダイアログボックス
- 新規計測 ２／２ ダイアログボックス
- 対象資産の編集 ダイアログボックス
- ページ指定 ダイアログボックス
- 印刷 ダイアログボックス
- 印刷範囲指定 ダイアログボックス
- 計測オプション ダイアログボックス
- 表示オプション ダイアログボックス
- サーチパスオプション ダイアログボックス
- CSVファイルオプション ダイアログボックス
- 帳票オプション ダイアログボックス
- java自動識別オプション ダイアログボックス
- VB6.0自動識別オプション ダイアログボックス
- [複数指定]

新規計測 1／2 ダイアログボックス

言語種別及び、対象資産の指定方法を指定します。

項目	説明
[言語種別]	ドロップダウンリストボックスを開き、計測対象資産の言語を、「COBOL」、「C/C++」、「Java」、「IDL」、「Java自動生成ソース」、「HTML/JSP/JS/CSS」、「.NET(ASP/C#/VB)/HTML/JS/CSS」、「Visual Basic 6.0」より選択します。
[対象資産の指定方法]	「ファイル」を選択すると、ファイル単位で計測対象資産を選択できます。 「フォルダ」を選択した場合はフォルダ単位で選択できます。サブフォルダを含めて検索するかどうかは、計測オプションで設定してください。
[文字コードの指定]	計測対象資産の文字コードを「ShiftJIS」、「UNICODE」より選択します。 「UNICODE」では、UTF-8およびBOM(Byte Order Mark)ありのUCS2（ビッグエンディアンまたはリトルエンディアン）に対応しています。

=> 新規計測 2/2 ダイアログボックス

新規計測 2／2 ダイアログボックス

計測対象資産を指定します。[新規計測 1／2]ダイアログボックスの[対象資産の指定方法]での選択により、[ファイルによる対象資産の指定]または、[フォルダによる対象資産の指定]画面が表示されます。

［ファイルによる対象資産の指定］の場合

項目	説明
[登録ファイル]	登録済みのファイル名（ファイル名とフォルダ名）の一覧が表示されます。複数選択が可能です。（削除で使用）
[追加]	[追加]ボタンを押下すると[ファイルの指定]ダイアログボックスが表示されます。そこでファイルを指定し、[開く]ボタンを押下すると[登録ファイル]にファイル名が登録されます。複数選択も可能です。 各言語毎に以下の拡張子のファイルを指定可能です。 COBOL ： (*.CBL)，(*.COB)，(*.COBOL)，(*.SCOB)，(*.PCO)，(*.*) C/C++ ： (*.C)，(*.CPP)，(*.H)，(*.RC)，(*.*) Java ： (*.JAVA)，(*.SQLJ)，(*.*) IDL ： (*.IDL)，(*.*) HTML/JSP/JS/CSS ： (*.HTML)，(*.HTM)，(*.JSP)，(*.JSPF)，(*.TAG)，(*.TAGX)，(*.TAGF)，(*.JS)，(*.CSS)， (*.*) .NET(ASP/C#/VB)/HTML/JS/CSS ： (*.ASP)，(*.ASPX)，(*.CS)，(*.VB)，(*.VBS)，(*.HTML)，(*.HTM)，(*.JS)，(*.CSS)， (*.*)

	Visual Basic 6.0 ： (*.FRM)，(*.BAS)，(*.CLS)，(*.CTL)，(*.PAG)，(*.DSR)，　(*.*)
[削除]	[登録ファイル]で削除するファイル名をクリックし、[削除]ボタンを押します。複数選択も可能です。

［フォルダによる対象資産の指定］の場合

項目	説明
[フォルダ]	[参照]ボタン押下による参照設定したフォルダ名の表示。直接手入力することも可能です。
[フォルダリスト]	登録済みのフォルダ名の一覧が表示されます。ダブルクリックにより、選択したフォルダ名が[フォルダ]エディットボックスに表示されます。複数選択が可能です。（削除で使用）
[参照]	フォルダ名を参照したい場合、[参照]ボタンを押します。[フォルダの参照]ダイアログボックスを表示します。
[追加]	[フォルダ]エディットボックスにフォルダ名が表示されている状態で[追加]ボタンを押下します。 計測オプションで[サブフォルダを検索する]を有効にした場合 既に上位フォルダが登録されている状態でその配下の下位フォルダを登録することはできません。 既に下位フォルダが登録されている状態（複数の場合あり）でその上位にあたるフォルダを登録することはできません。

[削除]	[フォルダリスト]で削除するフォルダ名をクリックし、[削除]ボタンを押します。複数選択も可能です。

対象資産の編集方法

指定した対象資産の編集を行います。

以下の場合、対象資産の編集を行って、全計測または変更部分のみ計測を行います。

1. 新規計測において指定した対象資産一覧を再編集したい。
2. 計測結果ファイル読み込み後の対象資産一覧の編集をしたい。
3. 計測後の対象資産一覧の編集をしたい。

対象資産を編集するには

1. [ファイル]メニューの[対象資産の編集]コマンドを選びます。
2. 対象資産の編集ダイアログボックスで必要項目の設定をします。
3. 全ての指定が済んだら、[OK]ボタンを押下します。
   この後、[全計測]または[変更部分のみ計測]により、計測処理が開始され、計測結果を画面に表示します。

注意　対象資産編集を行いOKボタンをクリックすると画面上に表示されている計測結果が未保存の場合でも無条件に破棄されます。

表示される画面（ファイルまたは、フォルダ）は、新規計測時に新規計測 1/2 ダイアログボックスの[対象資産の指定方法]で選択した方法に依存します。

対象資産の編集 ダイアログボックス

新規計測にて指定した対象資産一覧及び、計測結果ファイル読み込み後の対象資産一覧に追加/削除等の編集を行います。[新規計測]時にファイルによる指定をしていれば、[ファイルによる指定]画面が、フォルダによる指定をしていれば、[フォルダによる指定]画面が表示されます。

［ファイルによる指定］の場合

項目	説明
[登録ファイル]	登録済みのファイル名（ファイル名とフォルダ名）の一覧が表示されます。複数選択が可能です。
[追加]	[追加]ボタンを押下すると[ファイルの指定]ダイアログボックスが表示されます。そこでファイルを指定し、[開く]ボタンを押下すると[登録ファイル]にファイル名が登録されます。複数選択も可能です。 各言語毎に以下の拡張子のファイルを指定可能です。 COBOL ： (*.CBL), (*.COB), (*.COBOL), (*.SCOB), (*.PCO), (*.*) C/C++ ： (*.C), (*.CPP), (*.H), (*.RC) , (*.*) Java ： (*.JAVA), (*.SQLJ) , (*.*) IDL ： (*.IDL) , (*.*) Java自動生成ソース：(*.JAVA), (*.SQLJ) , (*.*) HTML ： (*.HTML), (*.HTM), (*.JSP), (*.JSPF), (*.TAG), (*.TAGX), (*.TAGF), (*.JS), (*.CSS), (*.*) ASP ： (*.ASP), (*.ASPX), (*.CS), (*.VB), (*.VBS), (*.HTML), (*.HTM), (*.JS), (*.CSS), (*.*) VB6.0 ： (*.FRM), (*.BAS), (*.CLS), (*.CTL), (*.PAG), (*.DSR), (*.*)

[削除]	[登録ファイル]で削除するファイル名をクリックし、[削除]ボタンを押します。複数選択も可能です。

［フォルダによる指定］の場合

項目	説明
[フォルダ]	[参照]ボタン押下による参照設定したフォルダ名の表示。直接手入力することも可能です。
[フォルダリスト]	登録済みのフォルダ名の一覧が表示されます。ダブルクリックにより、選択したフォルダ名が[フォルダ]エディットボックスに表示されます。複数選択が可能です。
[参照]	フォルダ名を参照したい場合、[参照]ボタンを押します。[フォルダの参照]ダイアログボックスを表示します。
[追加]	[フォルダ]エディットボックスにフォルダ名が表示されている状態で[追加]ボタンを押下します。 計測オプションで[サブフォルダを検索する]を有効にした場合 既に上位フォルダが登録されている状態でその配下の下位フォルダを登録することはできません。 既に下位フォルダが登録されている状態（複数の場合あり）でその上位にあたるフォルダを登録することはできません。
[削除]	[フォルダリスト]で削除するフォルダ名をクリックし、[削除]ボタンを押します。複数選択も可能です。

注意　対象資産の編集を行いOKボタンをクリックすると画面上に表示されている計測結果が未保存の場合でも無条件に破棄されます。

計測結果ファイルの読み込み方法

計測、表示、編集対象の計測結果ファイルを読み込みます。

ファイルを読み込むには

1. [ファイル]メニューの[読み込み]コマンドを選びます。
2. [ファイルを開く]ダイアログボックスより、計測結果ファイルを選択し[OK]ボタンを押下します。
3. 計測結果を画面に表示します。

計測結果ファイルの保存方法

 計測結果ファイルの保存方法(上書き保存)

 計測結果ファイルの保存方法(名前をつけて保存)

計測結果ファイルの保存方法(上書き保存)

再計測に於いて、元の計測結果ファイルを更新して保存します。

ファイルを保存するには

1. [ファイル]メニューの[上書き保存]コマンドを選びます。
2. 計測結果ファイルを新しい計測結果で上書き更新します。

 関連項目

計測結果ファイルの保存方法(名前をつけて保存)

新規計測の保存及び、既存の計測結果ファイル名の変更による保存を行います。

ファイルを保存するには

1. [ファイル]メニューの[名前を付けて保存]コマンドを選びます。
2. [ファイルの保存]ダイアログボックスより、計測結果ファイル名を指定し[OK]ボタンを押下します。
3. 指定したファイル名で計測結果ファイルが作成されます。

 関連項目

CSV形式ファイル作成

 CSV形式ファイル作成

 CSV形式ファイル作成(選択範囲のみ)

CSV形式ファイル作成

計測結果をCSV形式のファイルに保存します。計測結果の表示方式によって、保存されるフォーマットが異なります。

- 帳票形式時のCSV形式ファイル
- リストビュー形式時のCSV形式ファイル

ファイルを保存するには

1. [ファイル]メニューの[CSV形式ファイルの作成]コマンドを選びます。
2. [ファイルの保存]ダイアログボックスより、CSV形式ファイル名を指定し[OK]ボタンを押下します。
3. 指定したファイル名でCSV形式ファイルが作成されます。

 関連項目

CSV形式ファイル作成(選択範囲のみ)

選択されている計測結果のみCSV形式のファイル（リストビュー形式）に保存します。

ファイルを保存するには

1. CSV形式ファイルへ出力する計測結果をリストビューより選択します（複数選択可能）。
2. [ファイル]メニューの[選択範囲のみCSV形式ファイルの作成]コマンドを選びます。
3. [ファイルの保存]ダイアログボックスより、CSV形式ファイル名を指定し[OK]ボタンを押下します。

4. 指定したファイル名でCSV形式ファイルが作成されます。

注意　計測結果をリストビュー形式で表示し、合計や平均の行のみを選択している場合は、このコマンドを実行できません。

注意　[選択範囲のみCSV形式ファイル作成]時の合計、平均値は、CSV形式ファイルに出力した情報のみを対象としています。

関連項目

CSV形式ファイル

 CSV形式ファイル（リストビュー形式）

 CSV形式ファイル（帳票形式）

CSV形式ファイル（リストビュー形式）

計測結果ファイルは通常STEPCOUNTERが読み書きする専用の形式で保存されますが、「CSV形式ファイルの作成」を行うことにより、「CSV(Comma Separated Value)形式」のファイルで保存することができます。ただし、作成されたCSV形式ファイルを再びSTEPCOUNTERで読み込むことはできません。基本的にリストビューで表示している情報をそのままの形式で出力します。

CSV形式ファイルの出力例

"***** SIMPLIA/MF-STEPCOUNTER ***** 2000年08月09日 09:29:18 [UNICODE]"

"プログラムステップ情報"
"ファイル", "総ステップ数",,"手書きステップ数",,"組込みステップ数",
"ファイル名", "有効", "計", "有効", "計", "有効", "計"
"CMFSTP32.cpp", 4014, 6164, 1613, 2073, 2401, 4091
"合計", 4014, 6164, 1613, 2073, 2401, 4091
"平均", 4014, 6164, 1613, 2073, 2401, 4091

"組込みメンバステップ情報(include)"
"ファイル", "ステップ数内訳",
"ファイル名", "有効","計"
"childfrm.h", 23, 53
"cmfstp32.h", 99, 161
"合計", 122, 214
"平均", 61, 107

タイトル行について

・[タイトル]、[作成日時]、[文字コード]を出力します。[文字コード]は新規計測時に指定した文字コード(ShiftJISまたはUNICODE)が出力されます。

明細行について

・リストビュー形式画面に表示されているデータのみを出力対象としています。よって[表示フォルダ]コンボボックスの表示フォルダ指定により、表示データを絞り込んでいる場合、そのデータだけが出力対象となります。

出力項目について

・[CSVファイル]オプションの[非表示の項目は出力しない]を指定するとリストビュー形式画面で非表示としている項目の出力が抑止されます。

・各項目の詳細については、計測結果の説明を参照ください。

合計,平均行について

・[選択範囲のみCSV形式ファイル作成]時の合計、平均値は、CSV形式ファイルに出力した情報のみを対象としています。[CSVファイル]オプションの[合計，平均行は出力しない]を指定していない場合に出力されます。

CSV形式ファイル（帳票形式）

計測結果ファイルは通常STEPCOUNTERが読み書きする専用の形式で保存されますが、「CSV形式ファイルの作成」を行うことにより、「CSV(Comma Separated Value)形式」のファイルで保存することができます。ただし、作成されたCSV形式ファイルを再びSTEPCOUNTERで読み込むことはできません。また、帳票イメージでの一部の内容のみ出力されます。

CSV形式ファイルの出力形式

0,	"c:¥test¥",	"PROG0001",	318,	11,	0,	519,	27,	0,	0
0,	"c:¥test¥",	"PROG0002",	613,	24,	0,	768,	33,	0,	0
:									
2,	"c:¥test¥",	"ZCOPY01",	86,	5,	0,	0,	0,	0,	0
2,	"c:¥test¥",	"ZCOPY02",	56,	8,	0,	0,	0,	0,	0
:									
1,	"c:¥test¥",	"ZINC000",	46,	3,	0,	0,	0,	0,	0
1,	"c:¥test¥",	"ZINC001",	81,	3,	0,	0,	0,	0,	0
↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)

№	出力項目	説明
(1)	計測ファイル種別	0:C/C++ソースファイル、およびCOBOLソースファイル 1:C/C++インクルードファイル、およびCOBOLインクルードファイル 2:COBOLコピー登録集ファイル 4:リソースインクルードファイル 5:Java/SQLJソースファイル 6:IDLソースファイル 7:HTMLソースファイル 8:JSPソースファイル 9:JSソースファイル 10:CSSソースファイル 11:ASP.NETソースファイル 12:C#ソースファイル 13:VB.NETソースファイル 14:VBScriptソースファイル 15:Visual Basic 6.0ソースファイル

(2)	フォルダ名	帳票オプションの[ファイルパス名を出力する]が指定されている場合に出力　指定していない場合は空欄　ダブルクォート(")でくくる
(3)	計測ファイル名	計測ファイル名　ダブルクォート(")でくくる
(4)	宣言部実ステップ数	COBOL宣言部の実ステップ数 C/C++,Java/SQLJ,IDL,HTML,ASP,VB6.0の場合は 0
(5)	宣言部注釈ステップ数	COBOL宣言部の注釈ステップ数 C/C++,Java/SQLJ,IDL,HTML,ASP,VB6.0の場合は 0
(6)	宣言部埋め込みSQLステップ数	COBOL宣言部の埋め込みSQLステップ数 C/C++,Java/SQLJ,IDL,HTML,ASP,VB6.0の場合は 0
(7)	実行部実ステップ数	COBOL実行部の実ステップ数 C/C++,Java/SQLJ,IDL,HTML,ASP,VB6.0の場合は 実ステップ数
(8)	実行部注釈ステップ数	COBOL実行部の注釈ステップ数 C/C++,Java/SQLJ,IDL,HTML,ASP,VB6.0の場合は 注釈ステップ数(Javadocコメント含む)
(9)	JavaDocコメントステップ数	Java/SQLJのJavadocコメントのステップ数 COBOL,C/C++,IDL,HTML,ASP,VB6.0の場合は 0
(10)	実行部埋め込みSQLステップ数	COBOL実行部の埋め込みSQLステップ数 C/C++,SQLJの場合は 埋め込みSQLステップ数 Java,IDL,HTML,ASP,VB6.0の場合は 0

注意　計測オプションの[ドキュメント／計測種別]の指定に関わらず、出力されるのはプログラムステップ情報のみです。

注意　SQLステップ数については帳票オプションの[SQLステップ数情報を出力する]の指定に関わらず、常に出力されます。

計測結果の印刷方法

印刷に関する情報設定を行い、プリンタへの出力を行います。

計測結果を印刷するには

1. [ファイル]メニューの[印刷]コマンドを選びます。
2. 印刷 ダイアログボックスにて必要項目の設定をし、[OK]ボタンを押下します。
3. 計測結果がプリンタに印刷されます。

項目	内容
使用するプリンタ	現在選択されているプリンタを表示します。
ドキュメント種別／印刷範囲	印刷するドキュメント種別の選択およびドキュメントごとの印刷範囲を表示します。
印刷範囲	ドキュメントごとの印刷範囲を指定します。(指定ボタン押下時)
部数	ドキュメントごとの印刷部数を指定します。(指定ボタン押下時)
プリンタの設定	プリンタの設定を行います。

注意　プリンタ固有の設定についてはプリンタの取扱説明書を参照し、プリンタの能力にあった設定を行ってください。
計測結果を帳票形式で表示している場合のみ、このコマンドを実行できます。

印刷 ダイアログボックス

印刷に関する情報を設定します。

項目	説明
[使用するプリンタ]	現在選択されているプリンタを表示します。
[ドキュメント種別／印刷範囲]	印刷するドキュメント種別の選択およびドキュメントごとの印刷範囲を表示します。 ● ドキュメントを印刷対象としたい場合 印刷したいドキュメント種別名をクリックし、チェックマークを表示します。 ● 印刷範囲を変更したい場合 ドキュメント毎の[指定]ボタンを押下します。印刷範囲指定ダイアログボックスが表示されます。 ● 印刷部数を変更したい場合 ドキュメント毎の[指定]ボタンを押下します。印刷範囲指定ダイアログボックスが表示されます。
[プリンタの設定]	プリンタの変更／設定を行います。

印刷範囲指定 ダイアログボックス

印刷ページ範囲及び、印刷部数を設定します。

項目	説明
[ドキュメント]	設定対象のドキュメントを表示します。
[印刷範囲]	[全ページ]、[ページ指定]のどちらかを選択します。 [ページ指定]を選択した場合、[開始ページ]、[終了ページ] エディットボックスに値を設定します。両方に"0"を指定すると全ページが印刷されます。 [終了ページ]に"0"を指定すると最終ページまで印刷されます。 [終了ページ]に"0"以外のページを指定した時は、[開始ページ]に"0"以外のページ数を指定してください。
[部数]	印刷部数を[部数]エディットボックスに設定します。

ステップ数の計測方法(全計測)

新規計測及び、全対象ファイルの再計測を実行し、計測結果の表示を行います。

全計測するには

1. [計測]メニューの[全計測]コマンドを選びます。
2. 計測処理が開始され、計測結果を画面に表示します。

ステップ数の計測方法(変更部分のみ計測)

[対象資産の編集]処理にて対象資産の追加を行った部分に対してのみ計測し、計測結果の表示を行います。対象資産追加が行われず、オプションの変更のみが行われた場合、再計測しないで変更後のオプションでの計測結果再表示のみが実行されます。

変更部分の計測をするには

1. [計測]メニューの[変更部分のみ計測]コマンドを選びます。
2. 計測処理が開始され、計測結果を画面に表示します。

注意 以下の場合、変更部分のみ計測を行うことができません。

1. 新規計測の場合
2. 対象資産の編集で対象資産の削除を行った場合
3. サーチパスオプションでサーチパスの追加/削除を行った場合
4. 計測オプションの「組込みメンバを計測する」を変更した場合
5. 計測オプションの「埋め込みSQLを計測する」を変更した場合
6. 計測オプションの「サブフォルダも検索する」を変更した場合

計測オプションの設定

出力する計測結果の種類および、計測方法について設定します。

設定するには

1. [オプション]メニューの[計測オプション]コマンドを選びます。
2. 計測オプションダイアログボックスで必要項目の設定をします。

項目	内容
プログラムステップ情報を出力する	プログラムソースのステップ数を出力するかを指定します。
組込みステップ情報を出力する	プログラムソースより展開されている組込みメンバのステップ数を出力するかを指定します。
組込みメンバを計測する	組込みメンバのステップを計測するかを指定します。
埋め込みSQLを計測する	埋め込みSQLのステップを分けて計測するかを指定します。
サブフォルダも検索する	対象資産を指定する際、フォルダによる指定で行う場合、サブフォルダ配下に存在するファイルも計測対象とするかを指定します。　（注１）
プログラムID単位に計測する	COBOL言語種別で計測する際、プログラムID単位に計測するかを指定します。１つのソースファイル中に複数のプログラムが含まれている場合、副プログラムも計測します。
フォルダ指定の場合、ヘッダーファイル(*.h)も計測する	C/C++言語種別で計測する際、ヘッダーファイルもプログラムステップ情報として計測するかを指定します。（注２）

注１）計測オプション「サブフォルダも検索する」はプログラムステップの計測のみに適用され、組込みメンバを計測する場合のサーチパスは指定したフォルダのみ参照します。

注２）計測オプション「組込みメンバを計測する」が有効となっている場合、この機能は使用できません。ヘッダーファイルは組込み情報として計測されます。

注意　変更を計測結果に反映するためには、「全計測」または、「変更部分のみ計測」を実行する必要があります。ただし「組込みメンバを計測する」「埋め込みSQLを計測する」「サブフォルダも検索する」「プログラムID単位に計測する」「フォルダ指定の場合、ヘッダーファイル(*.h)も計測する」変更時は、「全計測」を実行する必要があります。

計測オプション ダイアログボックス

計測オプションは、出力する計測結果の種類および、計測方法について設定します。

項目	説明
[ドキュメント／計測種別]	1つ以上の選択が必要です。 ・プログラムステップ情報を出力する場合 　[プログラムステップ情報を出力する]をクリックし、チェックマークを表示します。 ・組込みメンバステップ情報を出力する場合 　[組込みステップ情報を出力する]をクリックし、チェックマークを表示します。 **注意** [組込みステップ情報を出力する]のみを選択して、htmlやjavaソース等、組込みメンバステップ情報の出力のないソースを計測すると、

	自動的に[プログラムステップ数を出力する]にチェックが入ります。
[計測方法]	・組込みメンバを計測する場合 　[組込みメンバを計測する]をクリックし、チェックマークを表示します。 ・埋め込みSQLを計測する場合 　[埋め込みSQLを計測する]をクリックし、チェックマークを表示します。 ・サブフォルダも検索する場合 　[サブフォルダも検索する]をクリックし、チェックマークを表示します。
[COBOLの計測方法]	・COBOL言語種別の計測で、プログラムID単位に計測する場合 　[プログラムID単位に計測する]をクリックし、チェックマークを表示します。
[C/C++の計測方法]	・C/C++言語種別の計測で、ヘッダーファイル(*.h)もプログラム情報として計測する場合 　[フォルダ指定の場合、ヘッダーファイル(*.h)も計測する]をクリックし、チェックマークを表示します。 注意 [組込みメンバを計測する]を選択すると、自動的に[フォルダ指定の場合、ヘッダーファイル(*.h)も計測する]のチェックが外され、選択できなくなります。

表示オプションの設定

画面の表示形式及びリストビュー形式画面に関する情報を指定します。

設定するには

1. [オプション]メニューの[表示オプション]コマンドを選びます。
2. 表示オプションダイアログボックスで必要項目の設定をします。

項目	内容
初期画面表示	計測結果を帳票形式／リストビュー形式のどちらで表示するかを指定します。
ファイル情報を表示	ファイル情報の種類、サイズ、更新日付の初期表示／非表示を指定します。
SQLステップ数を表示	SQL情報のSQL、ETCの初期表示／非表示を指定します。
SQLステップ率を表示	SQL情報のSQL率の初期表示／非表示を指定します。
注釈ステップ数を表示	注釈ステップ数の初期表示／非表示を指定します。
注釈率を表示	注釈率、Javadoc率の初期表示／非表示を指定します。
組込み情報を表示	プログラム情報では、組込み形態別項目、組込みメンバ情報では、組込み情報の初期表示／非表示を指定します。
フォルダ名を表示	フォルダ名の初期表示／非表示を指定します。
注釈率下限値	注釈率の下限値を設定します。設定値以下の注釈率を赤色表示します。

注意 変更を計測結果に反映するためには、「全計測」または、「変更部分のみ計測」を実行する必要があります。特に計測を行わずに表示形式のみを変更したい場合は、「変更部分のみ計測」を実行してください。

表示オプション ダイアログボックス

画面の表示形式及びリストビュー形式画面に関する情報を指定します。

項目	説明
[初期画面表示]	・帳票形式で計測結果を出力する場合 [帳票形式]をクリックし、チェックマークを表示します。 ・リストビュー形式で計測結果を出力する場合 [リストビュー形式]をクリックし、チェックマークを表示します。

<table>
<tr><td>[リストビュー形式]</td><td>・ファイル情報を表示する場合

[ファイル情報を表示]をクリックし、チェックマークを表示します。

・SQLステップ数を表示する場合

[SQLステップ数を表示]をクリックし、チェックマークを表示します。

・SQLステップ率を表示する場合

[SQLステップ率を表示]をクリックし、チェックマークを表示します。

・注釈ステップ数を表示する場合

[注釈ステップ数を表示]をクリックし、チェックマークを表示します。

・注釈率を表示する場合

[注釈率を表示]をクリックし、チェックマークを表示します。

・組込み情報を表示する場合

[組込み情報を表示]をクリックし、チェックマークを表示します。

・フォルダ名を表示する場合

[フォルダ名を表示]をクリックし、チェックマークを表示します。

・注釈率下限値を変更する場合

[注釈率下限値]の▼ボタンをクリックし、ドロップダウンリストから設定したい値を選択します。</td></tr>
</table>

サーチパスオプションの設定

ソースプログラムで展開されている、組込みファイルが存在するフォルダのパスを、最大10件まで指定できます。

設定するには

1. [オプション]メニューの[サーチパスの設定]コマンドを選びます。
2. サーチパスの設定ダイアログボックスで必要項目の設定をします。

項目	内容
フォルダ	追加対象のフォルダ名を表示／入力します。
フォルダリスト	登録済みのフォルダ名の一覧が表示されます。
参照	フォルダ名を参照します。
追加	[フォルダ]に表示されているフォルダ名を追加登録します。
削除	登録済のフォルダ名を登録解除します。

注意　・変更を計測結果に反映するためには、「全計測」を実行する必要があります。
・同一フォルダの重複登録はできません。
・計測時の検索順序は、登録順のとおりです。
・計測対象ファイルの存在するフォルダは、サーチパスに登録しなくても自動的に検索します。
・検索の優先順位は、サーチパスで登録したフォルダ・対象資産ファイルの存在するフォルダの順です。
・サーチパスはサブフォルダを検索する事はせず、指定したフォルダのみ検索します。

サーチパス ダイアログボックス

サーチパスに関する情報を指定します。

項目	説明
[フォルダ]	[参照]ボタン押下による参照設定したフォルダ名が表示されます。直接手入力することも可能です。 [追加]ボタンによりフォルダリストに追加してください。
[フォルダリスト]	登録済みのフォルダ名の一覧が表示されます。複数選択が可能です。(フォルダパスは、最大10件まで登録できます。) ダブルクリックにより、選択したフォルダ名が[フォルダ]エディットボックスに表示されます。この状態で[参照]ボタンを押下すると、[フォルダの参照]ダイアログボックスでそのフォルダを初期表示します。
[参照]	フォルダ名を参照したい場合、[参照]ボタンを押します。[フォルダの参照]ダイアログボックスを表示します。
[追加]	[フォルダ]エディットボックスにフォルダ名が表示されている状態で[追加]ボタンを押下すると、サーチパスとして[フォルダリスト]に追加されます。
[削除]	[フォルダリスト]で削除するフォルダ名をクリックし、[削除]ボタンを押します。複数選択も可能です。

CSVファイルオプションの設定

CSV形式ファイルの出力に関する情報を指定します。

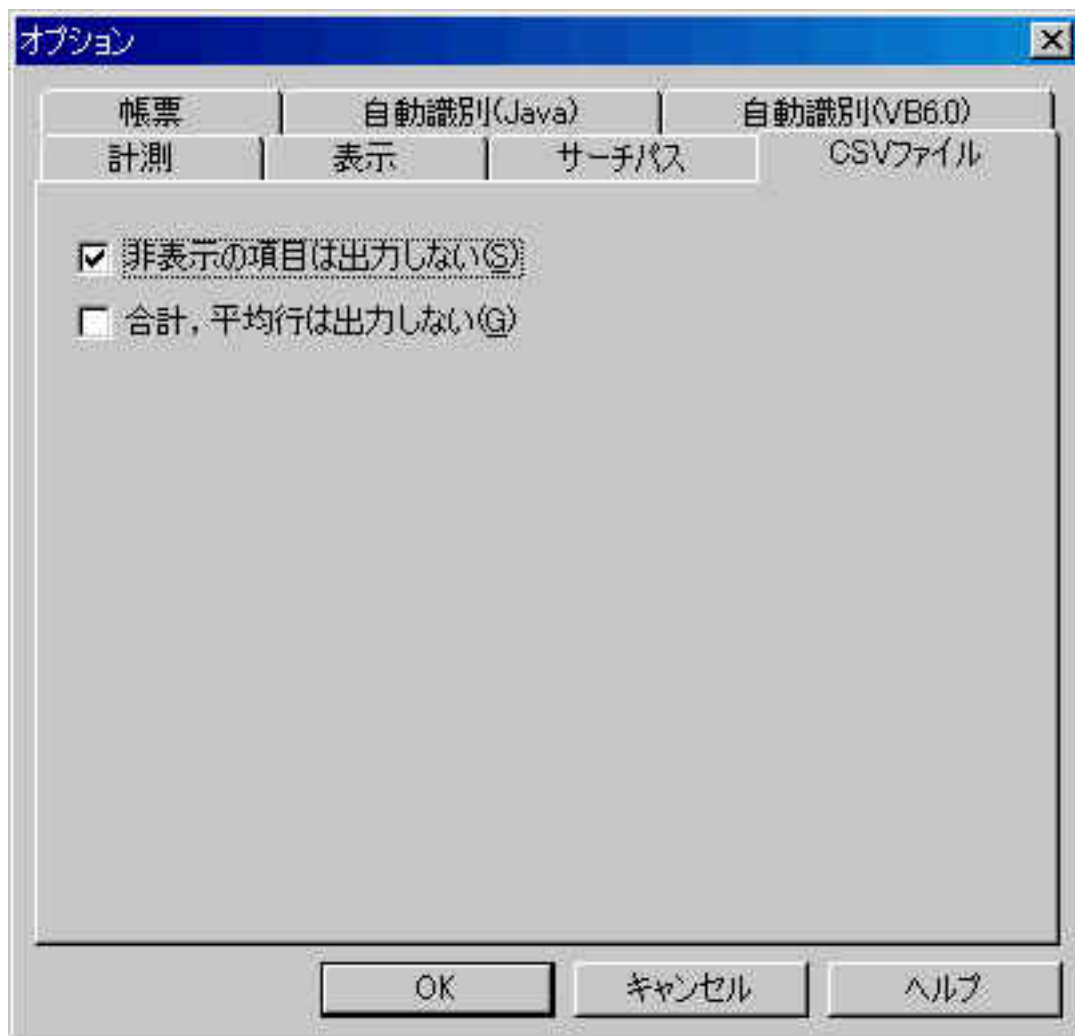
設定するには

1. [オプション]メニューの[CSVファイルオプションの設定]コマンドを選びます。
2. CSVファイルオプションダイアログボックスで必要項目の設定をします。

項目	内容
非表示の項目は出力しない	非表示としている項目を出力対象とするかを指定します。
合計，平均行は出力しない	合計、平均行を出力するかを指定します。

注意　計測結果をリストビュー形式で表示している場合のみ、有効となります。

CSVファイルオプション ダイアログボックス

画面の表示形式及びリストビュー形式画面に関する情報を指定します。

項目	説明
[非表示の項目は出力しない]	・非表示項目をCSV形式ファイルに出力しない場合 [非表示の項目は出力しない]をクリックし、チェックマークを表示します。
[合計，平均行は出力しない]	・合計、平均行をCSV形式ファイルに出力しない場合 [合計，平均行は出力しない]をクリックし、チェックマークを表示します。

帳票オプションの設定

帳票形式画面及び印刷帳票に関する情報を指定します。

設定するには

1. [オプション]メニューの[帳票オプション]コマンドを選びます。
2. 帳票オプションダイアログボックスで必要項目の設定をします。

項目	内容
ファイルパス名を出力する	フォルダ名を出力するかを指定します。
SQLステップ情報を出力する	[計測]オプションの[埋め込みSQLを計測する]に関係なくSQLステップ情報を出力するかを指定します。 [埋め込みSQLを計測する]にチェックが入っていない場合は0を出力します。
ファイルの更新日付を出力する	ファイルの更新日付を出力するかを指定します。
注記	帳票に印刷する注記を設定します（30バイトまで）。

注意 「注記」以外の項目は、変更を計測結果に反映するために、「全計測」または、「変更部分のみ計測」を実行する必要があります。

帳票オプション ダイアログボックス

帳票形式画面及び印刷帳票に関する情報を指定します。

項目	説明
[ファイルパス名を出力する]	・フォルダ名を出力する場合 [ファイルパス名を出力する]をクリックし、チェックマークを表示します。
[SQLステップ情報を出力する]	・SQLステップ情報を出力する場合 [SQLステップ情報を出力する]をクリックし、チェックマークを表示します。
[ファイルの更新日付を出力する]	・ファイルの更新日付を出力する場合 [ファイルの更新日付を出力する]をクリックし、チェックマークを表示します。
[注記]	・注記を設定する場合 [注記]エディットボックスに文字列を30バイト以内で入力します。

Java自動識別オプションの設定

Java自動生成ソース計測時の手書き部分を識別する、手書き開始／終了識別子に関する情報を指定します。

ここで指定された識別子を含む行の間を手書き部として計測し、それ以外を自動生成部として計測します。
計測基準の詳細についてはJava自動生成ソース計測基準を参照してください。

設定するには

1. [オプション]メニューの[Java自動識別オプション]コマンドを選びます。
2. Java自動識別オプションダイアログボックスで必要項目の設定をします。

項目	説明
[計測中に識別子不整合エラーを表示する]	・計測中に識別子不整合エラーを表示する場合 [計測中に識別子不整合エラーを表示する]をクリックし、チェックマークを表示します。
[識別子]	登録済みの手書き開始識別子・手書き終了識別子・コメントが表示されます。複数選択が可能です。(識別子の組み合わせは、最大100件まで指定可能です。)
[取り込み]	識別情報が格納される識別子ファイルの内容を、［識別子］に取り込む場合、[取り込み]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。このとき、［識別子］に表示される登録済み識別子は、消去されます。
[書き出し]	［識別子］に表示されている登録済み識別子を識別子ファイルとして保存する場合、[書き出し]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。
［追加］	［識別子］に新しい識別子を追加する場合、［追加］ボタンを押します。識別文字列指定ダイアログを表示します。
［削除］	［識別子］で削除する識別子をクリックし、［削除］ボタンを押します。複数選択が可能です。

標準では以下の識別子が登録されています。

開始識別子	終了識別子	コメント
// >> 外部アジャスト対象の開始位置	// >> 外部アジャスト対象の終了位置	CBMBuilder
//>>ユーザ領域開始	//ユーザ領域終了<<	ユーザ領域保護ツール
<!-->>ユーザ領域開始	<!-ユーザ領域終了<<-->	ユーザ領域保護ツール
//******************** 変更可能領域 開始 ********************	//******************** 変更可能領域 終了 ********************	ユーザ領域保護ツール

注意　変更を計測結果に反映するためには、「全計測」または、「変更部分のみ計測」を実行する必要があります。

Java自動識別オプション ダイアログボックス

Java自動生成ソース計測時の手書き部分／自動生成部分の計測に関する情報を指定します。

項目	説明
[計測中に識別子不整合エラーを表示する]	・計測中に識別子不整合エラーを表示する場合 [計測中に識別子不整合エラーを表示する]をクリックし、チェックマークを表示します。
[識別子]	登録済み開始識別子・終了識別子・コメントが表示されます。複数選択が可能です。(識別子の組み合わせは、最大100件まで指定可能です。)
[取り込み]	識別情報が格納される識別子ファイルの内容を、［識別子］に取り込む場合、[取り込み]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。このとき、［識別子］に表示される登録済み識別子は、消去されます。
[書き出し]	［識別子］に表示されている登録済み識別子を識別子ファイルとして保存する場合、[書き出し]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。
［追加］	［識別子］に新しい識別子を追加する場合、［追加］ボタンを押します。識別文字列指定ダイアログを表示します。

［削除］	［識別子］で削除する識別子をクリックし、［削除］ボタンを押します。複数選択が可能です。

識別文字列指定ダイアログボックス

Java自動生成ソース計測時の手書き部分／自動生成部分を判定する識別子を指定します。

項目	説明
[開始文字列]	開始識別子を指定する。
[終了文字列]	終了識別子を指定する。
[コメント]	識別子に対するコメントを指定する。

VB6.0自動識別オプションの設定

Visual Basic 6.0ソース計測時の手書き部分を識別する、手書き開始／終了識別子に関する情報を指定します。

ここで指定された識別子を含む行の間を手書き部として計測し、それ以外を自動生成部として計測します。
計測基準の詳細についてはVB6.0自動生成ソース計測基準を参照してください。

設定するには

1. [オプション]メニューの[VB6.0自動識別オプション]コマンドを選びます。
2. VB6.0自動識別オプションダイアログボックスで必要項目の設定をします。

項目	説明
[計測中に識別子不整合エラーを表示する]	・計測中に識別子不整合エラーを表示する場合 [計測中に識別子不整合エラーを表示する]をクリックし、チェックマークを表示します。
[識別子]	登録済みの手書き開始識別子・手書き終了識別子・コメントが表示されます。複数選択が可能です。(識別子の組み合わせは、最大100件まで指定可能です。)
[取り込み]	識別情報が格納される識別子ファイルの内容を、［識別子］に取り込む場合、[取り込み]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。このとき、［識別子］に表示される登録済み識別子は、消去されます。
[書き出し]	［識別子］に表示されている登録済み識別子を識別子ファイルとして保存する場合、[書き出し]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。
［追加］	［識別子］に新しい識別子を追加する場合、［追加］ボタンを押します。識別文字列指定ダイアログを表示します。
［削除］	［識別子］で削除する識別子をクリックし、［削除］ボタンを押します。複数選択が可能です。

標準では以下の識別子が登録されています。

開始識別子	終了識別子	コメント
VERSION 5.00	Attribute VB_Exposed	VB6.0(*.frm)
VERSION 1.0 CLASS	Attribute VB_Exposed	VB6.0(*.cls等)

注意　標準で登録されているVB6.0用の識別子はマイクロソフト社の公開された仕様に基づいたものではありません。このため、正確に計測されない場合は識別子を編集／追加してください。

注意 変更を計測結果に反映するためには、「全計測」または、「変更部分のみ計測」を実行する必要があります。

VB6.0自動識別オプション ダイアログボックス

Visual Basic 6.0ソース計測時の手書き部分／自動生成部分の計測に関する情報を指定します。

項目	説明
[計測中に識別子不整合エラーを表示する]	・計測中に識別子不整合エラーを表示する場合 [計測中に識別子不整合エラーを表示する]をクリックし、チェックマークを表示します。
[識別子]	登録済み開始識別子・終了識別子・コメントが表示されます。複数選択が可能です。(識別子の組み合わせは、最大100件まで指定可能です。)
[取り込み]	識別情報が格納される識別子ファイルの内容を、［識別子］に取り込む場合、[取り込み]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。このとき、［識別子］に表示される登録済み識別子は、消去されます。
[書き出し]	［識別子］に表示されている登録済み識別子を識別子ファイルとして保存する場合、[書き出し]ボタンを押します。［ファイルの参照ダイアログボックス］を表示します。
［追加]	［識別子］に新しい識別子を追加する場合、［追加］ボタンを押します。識別文字列指定ダイアログを表示します。

［削除］	［識別子］で削除する識別子をクリックし、［削除］ボタンを押します。複数選択が可能です。

識別文字列指定ダイアログボックス

Visual Basic 6.0ソース計測時の手書き部分／自動生成部分を判定する識別子を指定します。

項目	説明
[開始文字列]	開始識別子を指定する。
[終了文字列]	終了識別子を指定する。
[コメント]	識別子に対するコメントを指定する。

[複数指定]

範囲指定するには

対象範囲の先頭項目をクリックし、シフトキーを押しながら、末尾項目をクリックします。または先頭項目付近の空白領域（ファイル名以外の領域）をクリックしてボタンを押したまま末尾項目までマウスポインタを上下にずらし、ボタンを放します。マウスを使用しない場合はシフトキーを押したまま上下矢印キーで項目移動することにより選択範囲を指定できます。

個別指定するには

コントロールキーを押しながら、項目をクリックします。キーボード操作のみでの個別指定はできません。

プログラム情報

- プログラムステップ情報（リストビュー形式）
- プログラムステップ情報（帳票形式）
- 組込みメンバステップ情報（リストビュー形式）
- 組込みメンバステップ情報（帳票形式）

プログラムステップ情報（帳票形式）	プログラムステップ情報（リストビュー形式）	組込みメンバステップ情報（帳票形式）	組込みメンバステップ情報（リストビュー形式）

プログラムステップ情報の出力レイアウト例（C/C++の帳票）を以下に示します。知りたい箇所をクリックしてください。

2000年08月08日作成　　1PAGE

プログラムステップ情報 [C/C++]　　　　　　　　注記 (　　　　　　　　　　　　　)

No.	プログラム名	ステップ数内訳			組込み形態別
		総ステップ	手書き	組込み	
E:¥work¥Mfstp¥SRC					
1	F5avscnt.c	有効 2,004(77%)	有効 1,497(78%)	有効 507(74%)	I 686(100%)
		(SQL) 0(0%)	(SQL) 0(0%)	(SQL) 0(0%)	R -----
		(ETC) 2,004(100%)	(ETC) 1,497(100%)	(ETC) 507(100%)	
	2000/05/26	注釈 611(23%)	注釈 432(22%)	注釈 179(26%)	
	09:11:58	計 2,615	計 1,929(74%)	計 686(26%)	
	小計	有効 2,004(77%)	有効 1,497(78%)	有効 507(74%)	I 686(100%)
		(SQL) 0(0%)	(SQL) 0(0%)	(SQL) 0(0%)	R 0(0%)
		(ETC) 2,004(100%)	(ETC) 1,497(100%)	(ETC) 507(100%)	
		注釈 611(23%)	注釈 432(22%)	注釈 179(26%)	
		計 2,615	計 1,929(74%)	計 686(26%)	
全体					
	合計	有効 2,004(77%)	有効 1,497(78%)	有効 507(74%)	I 686(100%)
		(SQL) 0(0%)	(SQL) 0(0%)	(SQL) 0(0%)	R 0(0%)
		(ETC) 2,004(100%)	(ETC) 1,497(100%)	(ETC) 507(100%)	
		注釈 611(23%)	注釈 432(22%)	注釈 179(26%)	
		計 2,615	計 1,929(74%)	計 686(26%)	
	平均	有効 2,004(77%)	有効 1,497(78%)	有効 507(74%)	I 686(100%)
		(SQL) 0(0%)	(SQL) 0(0%)	(SQL) 0(0%)	R 0(0%)
		(ETC) 2,004(100%)	(ETC) 1,497(100%)	(ETC) 507(100%)	
		注釈 611(23%)	注釈 432(22%)	注釈 179(26%)	
		計 2,615	計 1,929(74%)	計 686(26%)	

※ それぞれの言語種別により若干帳票イメージが異なります。

　関連項目

プログラムステップ情報（リストビュー形式）

プログラムステップ情報 （帳票形式）	プログラムステップ情報 （リストビュー形式）	組込みメンバステップ情報 （帳票形式）	組込みメンバステップ情報 （リストビュー形式）

プログラムステップ情報の出力レイアウト例（C/C++の帳票）を以下に示します。知りたい箇所をクリックしてください。

ﾌﾟﾛｸﾞﾗﾑｽﾃｯﾌﾟ情報 (C/C++)

ファイル				総ステップ数						
ファイル名	種類	サイズ	更新日付	有効	SQL	ETC	SQL率	注釈	注釈率	計
F5avscnt.c	C/C++	59,436	2000/05/26 09:11:58	2,004	0	2,004	0%	611	23%	2,615
合計		59,436		2,004	0	2,004	0%	611	23%	2,615
平均		59,436		2,004	0	2,004	0%	611	23%	2,615

ファイル	手書きステップ数							組込みステップ数							組込み形態別		フォルダ
ファイル名	有	S...	E...	S...	注	注	計	有	S...	E...	S...	注	注	計	inc	rc	フォルダ名
F5avscnt.c	1...	0	1...	0...	4...	2...	1...	5...	0	5...	0...	1...	2...	6...	686	-----	E:¥work¥Mfstp¥SRC
合計	1...	0	1...	0...	4...	2...	1...	5...	0	5...	0...	1...	2...	6...	686	-----	
平均	1...	0	1...	0...	4...	2...	1...	5...	0	5...	0...	1...	2...	6...	686	-----	

※ それぞれの言語種別により若干帳票イメージが異なります。

任意の行を選択して右クリックすることにより、ポップアップメニューが表示されて以下の操作が可能になります。

- 以下の表示／非表示の切り替え（計測言語種類に依存します）
  ファイル情報／SQLステップ数／SQLの比率／注釈ステップ数／注釈の比率／組込み形態別／フォルダ名
- クリップボードへのCSV形式でのコピー

 関連項目

組込みメンバステップ情報（帳票形式）

プログラムステップ情報（帳票形式）	プログラムステップ情報（リストビュー形式）	組込みメンバステップ情報（帳票形式）	組込みメンバステップ情報（リストビュー形式）

組込みメンバステップ情報の出力レイアウト例（C/C++の帳票）を以下に示します。知りたい箇所をクリックしてください。

2000年08月08日作成　　1PAGE

組込みメンバステップ情報 [include]　　　　注記 (　　　　)

No.	組込み メンバ名	ステップ数内訳	使用箇所	のべ総ステップ	使用回数	使用プログラム及び組込みメンバ名
E:¥work¥Mfstp¥SRC						
1	f5avscnt.h00 1996/10/04 14:08:06	有効 9(69%) (SQL) 0(0%) (ETC) 9(100%) 注釈 4(31%) 計 13	--	13	1	(P) F5avscnt.c
2	f5avsmai.h 2000/05/10 12:18:17	有効 498(74%) (SQL) 0(0%) (ETC) 498(100%) 注釈 175(26%) 計 673	--	673	1	(P) F5avscnt.c
	小計	有効 507(74%) (SQL) 0(0%) (ETC) 507(100%) 注釈 179(26%) 計 686				
全体						
	合計	有効 507(74%) (SQL) 0(0%) (ETC) 507(100%) 注釈 179(26%) 計 686				
	平均	有効 254(74%) (SQL) 0(0%) (ETC) 254(100%) 注釈 90(26%) 計 344				

※ それぞれの言語種別により若干帳票イメージが異なります。

関連項目

組込みメンバステップ情報（リストビュー形式）

プログラムステップ情報 （帳票形式）	プログラムステップ情報 （リストビュー形式）	組込みメンバステップ情報 （帳票形式）	組込みメンバステップ情報 （リストビュー形式）

組込みメンバステップ情報の出力例（C/C++の帳票）を以下に示します。知りたい箇所をクリックしてください。

組込みメンバステップ情報 (include)

ファイル				ステップ数内訳							組み込み情報				フォルダ
ファイル名	種類	サイズ	更新日付	有効	SQL	ETC	SQL率	注釈	注釈率	計	使用個所	のべ総ステップ	使用回数	使用プログラム...	フォルダ名
f5avscnt.h00	C/C++	389	1996/10/04 14:08:06	9	0	9	0%	4	31%	13	--	13	1	(P) F5avscnt.c	E:¥work¥Mfstp¥SRC
f5avsmai.h	C/C++	34,074	2000/05/10 12:18:17	498	0	498	0%	175	26%	673	--	673	1	(P) F5avscnt.c	E:¥work¥Mfstp¥SRC
合計		34,463		507	0	507	0%	179	26%	686					
平均		17,232		254	0	254	0%	90	26%	343					

※ それぞれの言語種別により若干帳票イメージが異なります。

 関連項目

計測基準

- COBOL計測基準
- C/C++計測基準
- Ｊａｖａ言語計測基準
- IDL言語計測基準
- Ｊａｖａ自動生成ソース計測基準
- HTML/JSP/JS/CSS計測基準
- .NET（ASP/C#/VB）/HTML/JS/CSS計測基準
- Visual Basic 6.0

COBOL計測基準

COBOL計測基準	C/C++計測基準	Java計測基準	IDL計測基準	Java自動生成ソース計測基準	HTML計測基準	.NET計測基準	VB6.0計測基準

COBOLの計測基準は以下のようになります。

注記判定

- ７カラム目が次のいずれかであるもの。

  - 「*」 一般注記行
  - 「/」 改ページ指定
  - 「D」 デバッグ行

- 先頭行が次のいずれかであるもの。

  - 「@OPTIONS」 翻訳オプション指定
  - 「PROCESS」 翻訳オプション指定
  - 「CBL」 翻訳オプション指定

- 埋め込みSQL文の注記。
- 空行。
- 行内注記 「*>」 から行末まで。

備考

- COBOLの自由形式の正書法には対応していません。
- .NETプログラミング機能及びオブジェクト指向プログラミング機能におけるプログラム構造には対応していません。

登録集判定

- 登録集（COPY）の判定は次のようになります。

  - COPY ・・・・・・

- 登録集（INCLUDE）の判定は次のようになります。

  - INCLUDE ・・・・・・

備考

・ 語 COPY、INCLUDEで拡張子を含めたファイル名を指定している場合、新規計測の方法で示した拡張子以外のファイルも計測されます。
・ 語 COPY、INCLUDEと登録集ファイル名の間には一つ以上の空白かタブが必要です。また、改行している場合は計測対象ファイルの制限となります。
・ 語 COPY、INCLUDEはレコードの先頭から300バイト以内に記述されていなければなりません。
・ 埋め込みSQLのインクルードファイル（EXEC SQL INCLUDE）や、埋め込みSQL中の登録集は組込みメンバの対象にはなりません。
・ AIMのDBを使用する際の 「SUBSCHEMA名」 は登録集として計測しません。

宣言部と実行部の判定

・ 宣言部の判定基準を以下に示します。

- ソースファイルの先頭を含め 「IDENTIFICATION DIVISION」 から「PROCEDURE DIVISION」 の直前まで。

・ 実行部の判定基準を以下に示します。

- 「PROCEDURE DIVISION」 よりソースファイル終了まで。

備考

・ 計測対象となるCOBOLプログラムはソースファイル中に見出し部の開始（語 IDENTIFICATION DIVISION）と手続き部の開始（語 PROCEDURE DIVISION）が記述されていなければなりません。どちらか一方あるいは両方が未記述やコピー句登録集ファイル内に記述されている場合、STEPCOUNTERは正常な計測ができません。

・ 計測オプション[プログラム単ID位に計測する]が選択されている場合、「IDENTIFICATION DIVISION」から次の「IDENTIFICATION DIVISION」の直前までを 1 プログラムとして、メインプログラムと副プログラムを計測します。
プログラムが入れ子になっている場合は、外側のプログラムのプログラム終わり見出しは、最後に検出したプログラムの計測結果としてステップ数が加算されます。
出力される計測結果に、ファイル名とプログラムID名が表示されます。

・ 計測オプション[プログラムID単位に計測する]が選択されていない場合、メインプログラムのみ計測されます。ただし、メインプログラムとして計測されるのは、副プログラムの「IDENTIFICATION DIVISION」より前までの行になります。出力される計測結果に、ファイル名のみが表示されます。

埋め込みSQL判定（COBOL）

ANSI規格に準拠しているSQL文法を示します。（ORACLE、RDB II、informixなど）

1. 宣言節

```
EXEC SQL BEGIN DECLARE SECTION END-EXEC ┐
01 ～┐  │
01 ～ ├ ホスト変数、標識変数の宣言 ├ 宣言も埋め込みSQL部として
01 ～┘  │ カウントします。（５ステップ）
EXEC SQL END DECLARE SECTION END-EXEC ┘
```

2. その他

```
EXEC SQL ┐
   ～     ├ 「EXEC SQL」から「END-EXEC」までを埋め込みSQL部として
END-EXEC ┘ カウントします。（３ステップ）
```

- 埋め込みSQL中で「-（ハイフン）」が２個連続して出現した場合、その行末まで注記とみなします。

- 語 EXECと語 SQLの間は一つ以上の空白かタブが必要です。また、改行している場合は計測対象ファイルの制限となります。
- 語 EXEC、SQLはレコードの先頭から300バイト以内に記述されていなければなりません。

計測基準例

判定	ソースプログラム記述(sample.cob)
	.....*....1....*....2....*....3....*....4....*....5....*....6....
コメント	000100 @OPTIONS QUOTE
コメント	000200* *
コメント	000300* ステップ数計測機能　COBOL ソース計測の例
コメント	000400* ALL Rights Reserved Copyright FUJITSU LIMITED 2004
コメント	000500* *
	000600 IDENTIFICATION DIVISION.
	000700 PROGRAM-ID. TEST01.
	000800 AUTHOR. FUJITSU LTD.
コメント	000900*
	001000 ENVIRONMENT DIVISION.
	001100 CONFIGURATION SECTION.
	001200 SOURCE-COMPUTER. FACOM.
	001300 OBJECT-COMPUTER. FACOM.
コメント	001400D WITH DEBUGGING MODE.
	001500 SPECIAL-NAMES.
	001600 SUBSCHEMA-NAME. "AAAAA".
	001700 INPUT-OUTPUT SECTION.
	001800 FILE-CONTROL.
コメント	001900* 表示ファイル
	002000 SELECT DSP-FILE ASSIGN TO DSPDD.
コメント	002100/
	002200 DATA DIVISION.

```
コメント    002300*  基底場所節
            002400 BASED-STORAGE          SECTION.
コメント    002500*  ファイル節
            002600 FILE                   SECTION.
            002700 FD DSP-FILE.
            002800     INCLUDE DSP01.
コメント    002900*  作業場所節
            003000 WORKING-STORAGE        SECTION.
コメント    003100*  ＲＤＢホスト変数
SQL文       003200 EXEC SQL BEGIN DECLARE SECTION END-EXEC.
SQL文       003300 01 SQL-USERID PIC X(20).
SQL文       003400 01 SQL-PASSWD PIC X(20).
SQL文       003500 EXEC SQL END DECLARE SECTION END-EXEC.
コメント    003600*  共通領域
            003700     COPY COMMWK1 REPLACING ==X== BY ==C==.
コメント    003800*  定数節
            003900     COPY CONST01 OF COMMDD.
            004000 LINKAGE                SECTION.
            004100 REPORT                 SECTION.
            004200 SCREEN                 SECTION.
コメント    004300/
            004400 PROCEDURE              DIVISION.
            004500 MAIN01                 SECTION.
SQL文       004600      EXEC SQL
SQL文       004700           CONNECT :SQL-USERID
SQL文       004800                   IDENTIFIED BY :SQL-PASSWD
コメント    004900                   -- "PASS-WORD"
SQL文       005000           END-EXEC.
コメント    005100
            005200      MOVE A TO B. *> Ａは～
コメント    005300                  *> Ｂは～
コメント    005400* 次のMOVE命令は３行としてカウントされる
            005500      MOVE
            005600           B TO
            005700           C.
コメント    005800
            005900      STOP RUN.
コメント    006000/
```

[プログラムID単位に計測する]が選択されている場合は以下TEST011として計測され、選択されていない場合は計測対象外となる。

```
            006100 IDENTIFICATION         DIVISION.
            006200 PROGRAM-ID.            TEST011.
コメント    006300*
            006400 END PROGRAM            TEST011.
            006500 END PROGRAM            TEST01.
            006600
```

COBOL計測基準	C/C++計測基準	Java計測基準	IDL計測基準	Java自動生成ソース計測基準	HTML計測基準	.NET計測基準	VB6.0計測基準

C/C++の計測基準は以下のようになります。この計測基準はRDB部分を除きリソースファイルにも適用されます。

注記判定

- 「/*」 と 「*/」 で括られたステップ。（C,C++）
- 「//」 から行末まで。（C++）
- 埋め込みSQL文の注記。（C）
- 空行。（C,C++）

インクルード判定

- #include <・・・・・・>
- #include "・・・・・・"
- #include ・・・・・
- rcinclude ・・・・・

備考

- #includeは小文字で記述されていなければなりません。
- 埋め込みSQLのインクルードファイル（EXEC SQL INCLUDE）は組込みメンバの対象にはなりません。
- 埋め込みSQL中のインクルード（#include等）は組込みメンバの対象にはなりません。

埋め込みSQL判定（C）

ANSI規格に準拠したSQL文法（ORACLE、RDBⅡ、informixなど）か、informix ESQL/Cに準拠している文法。

- 「EXEC SQL」 から 「；（セミコロン）」 まで。
- 「$」 から 「；（セミコロン）」 まで。
- 埋め込みSQL中で 「-（ハイフン）」 が２個連続して出現した場合、その行末まで注記とみなします。

備考

- 「EXEC SQL」は大文字で記述されていなければなりません。

・ 語 EXECと語 SQLの間は一つ以上の空白かタブが必要です。また、改行している場合は計測対象ファイルの制限となります。
・ C/C++の変数で"$"を変数名に含む場合、SQL行と見なされます。必要に応じてSQL計測をしないように計測オプションを変更してください。

計測基準例

```
判定        ソースプログラム記述(sample.c)
コメント    /* ステップ数計測機能　C 言語 ソース計測の例 */
コメント    /* ALL Rights Reserved Copyright FUJITSU LIMITED 2004 */
            #include <stdio.h>
            #include "pp.h"
コメント
SQL文       EXEC SQL BEGIN DECLARE SECTION;
SQL文            VARCHAR user[20];
コメント    /* VARCHAR user[20]; */
SQL文             long ten;
SQL文             char stname[14];
SQL文             char kouza _code[6];
SQL文             char kouza _name[24];
SQL文       EXEC SQL END DECLARE SECTION;
SQL文       EXEC SQL INCLUDE SQLCA;
コメント
            main()
            {
                  long I,j,max,ten_sv;
コメント
                  strcpy(user.arr,"SCOTT");     /*文字列の複写*/
                  user.len=strlen(user.ar);
                  strcpy(pass.arr,"TIGER");
                  pass.len=strlen(pass.arr);
SQL文             EXEC SQL CONNECT : user IDENTIFIED BY :pass;             --COMMENT
SQL文             EXEC SQL WHENEVER NOT FOUND GOTO NOT_FOUNTED;
                  while(1) {
SQL文                 EXEC SQL DECLARE TEN_CUR CURSOR FOR
SQL文                                 SELECT SEISEKI.TEN, STUDENT.STNAME
SQL文                                     FROM SEISEKI, STUDENT, KOZA
SQL文                                 WHERE SEISEKI.STNO=STUDENT.STNO            --COMMENT
コメント    --COMMENT
SQL文                                     AND SEISEKI.KOUZA_CODE=:kouza _code
SQL文                                     AND SEISEKI.KOUZA_code=:kouza _code
SQL文                             ORDER BY SEISEKI.KOUZA_CODE=:kouza _code;
コメント     /*            printf("¥n講座コードを入力してください(eで終了)");*/
                     scanf("%s",kouza_code);
```

Ｊａｖａ言語計測基準

Javaの計測基準は以下のようになります。

注記判定

- 「/*」 と 「*/」 で括られたステップ。
- 「//」 から行末まで。
- 「/**」 と 「*/」 で括られたステップ。(Javadocコメント)
- 埋め込みSQL文の注記。
- 空行。

埋め込みSQL判定（Java）

- 「#sql」 から 「；（セミコロン）」 まで。
  #sql *** *** ;
  #sql *** = { *** } ;
  #sql { *** } ;
  「{ }(括弧)」が無い時は、最初の「；（セミコロン）」 まで。
  「{ }」がある時は、最初の「}；」 まで。
  「{ }」のネストは無いものとする。
- 埋め込みSQL中で 「-（ハイフン）」 が２個連続して出現した場合、その行末まで注記とみなします。

計測基準例

判定	ソースプログラム記述(sample.java)
コメント	/*
コメント	** ステップ数計測機能　SQLJ ソース計測の例
コメント	** ALL Rights Reserved Copyright FUJITSU LIMITED 2004
コメント	/*
	import java.sql.*;
	import sqlj.runtime.ref.DefaultContext;　 //sqljランタイムクラスのimport
コメント	//sqljランタイムクラスのimport
	import oracle.sqlj.runtime.Oracle;
コメント	
SQL文	#sql iterator Cursor1(String empno, String firstnme);
SQL文	#sql iterator
SQL文	Cursor2(String);
コメント	
	public class sampleClass {
SQL文	#sql cursor1 = { SELECT empno,firstnme from employee };
コメント	
SQL文	#sql { begin
SQL文	execute immediate
SQL文	`insert into ` ||:which_table ||
SQL文	`(ename, empno, sal) values(:1, :2, :3)'
コメント	-- note: PL/SQL rule is table | col name cannot be
コメント	-- a bind argument in USING
コメント	-- also,binds are by position except in dynamic PL/SQL blocks
SQL文	using :ename, :empno, :sal;
SQL文	end;
SQL文	};
コメント	
javadoc	/**
javadoc	更新を行うスタティックメソッド
javadoc	*/
	private static void staticUpdateReturning(int empno, double newSal)
	throws SQLException {
	System.out.println("static update-returning for empno " + empno);
	String ename;
SQL文	#sql { begin
SQL文	update emp set sal = :newSal
SQL文	where empno = :empno
SQL文	returning ename into :OUT ename; -- :OUT is for SQLJ bind

IDL言語計測基準

COBOL計測基準	C/C++計測基準	Java計測基準	IDL計測基準	Java自動生成ソース計測基準	HTML計測基準	.NET計測基準	VB6.0計測基準

IDLの計測基準は以下のようになります。

注記判定

- 「/*」と「*/」で括られたステップ。
- 「//」から行末まで。
- 空行。

計測基準例

判定	ソースプログラム記述(sample.idl)
コメント	/*
コメント	** ステップ数計測機能　IDL ソース計測の例
コメント	** ALL Rights Reserved Copyright FUJITSU LIMITED 2004
コメント	*/
	module Module1 {
コメント	
コメント	// インタフェース宣言
	interface Func1 {
	typedef long otype;
	exception FuncException { //例外宣言
コメント	//例外宣言
	string reason;
	};
コメント	
	otype Open(in string name)
	raises(FuncException);
	readonly attribute long data;
	typedef Object FuncObject;
	}
	}
	[EOF]

Ｊａｖａ自動生成ソース計測基準

COBOL計測基準	C/C++計測基準	Java計測基準	IDL計測基準	Java自動生成ソース計測基準	HTML計測基準	.NET計測基準	VB6.0計測基準

Java自動生成ソースの計測基準は以下のようになります。

注記判定・埋め込みSQL判定（Java）

・Java言語計測基準に準拠します。

手書き・自動生成部分の判定

・手書き・自動生成部分は、自動識別子オプションダイアログで設定する、開始識別子・終了識別子をもとに判定され、開始識別子の直後の行から終了識別子の直前の行までを手書き部分とみなし、それ以外の部分を自動生成部分とみなし計測します。
・自動識別子は前方一致で検索されます。
・以下の識別子は、判定の対象外となるため、注意が必要です。
　－１つの開始識別子・終了識別子の範囲内に存在する、入れ子状態の開始識別子・終了識別子
　－開始識別子・終了識別子の組み合わせが不当である場合の終了識別子
　－開始識別子が存在しない終了識別子
・自動識別子が1対も無いソースの計測結果は、全て自動生成ソース（組込みステップ数）として計測されます。

備考

・ 計測結果出力時は、自動生成部分を組込みステップ情報として出力します。
・ 標準で登録されている識別子については、Java自動識別オプションの設定を参照ください。

計測基準例

判定	ソースプログラム記述(生成sample.java)
組込み/コメント	/*
組込み/コメント	** ステップ数計測機能　SQLJ 手書き・自動生成部分判定の例
組込み/コメント	** ALL Rights Reserved Copyright FUJITSU LIMITED 2004
組込み/コメント	*/
組込み/有効	import java.sql.*;
組込み/有効	import sqlj.runtime.ref.DefaultContext;　 //sqljランタイムクラスのimport
組込み/コメント	//sqljランタイムクラスのimport
組込み/有効	import oracle.sqlj.runtime.Oracle;
組込み/コメント	
組込み/SQL文	#sql iterator Cursor1(String empno, String firstnme);
組込み/SQL文	#sql iterator
組込み/SQL文	Cursor2(String);
組込み/コメント	
組込み/有効	public class sampleClass {
組込み/SQL文	#sql cursor1 = { SELECT empno,firstnme from employee };
組込み/コメント	
組込み/SQL文	#sql { begin
組込み/SQL文	execute immediate
組込み/SQL文	`insert into ` ||:which_table ||
組込み/SQL文	`(ename, empno, sal) values(:1, :2, :3)'
組込み/コメント	-- note: PL/SQL rule is table	col name cannot be
組込み/コメント	-- a bind argument in USING
組込み/コメント	-- also,binds are by position except in dynamic PL/SQL blocks
組込み/SQL文	using :ename, :empno, :sal;
組込み/SQL文	end;
組込み/SQL文	};
組込み/コメント	//>>ユーザ領域開始
手書き/javadoc	/**
手書き/javadoc	更新を行うスタティックメソッド
手書き/javadoc	*/
手書き/有効	private static void staticUpdateReturning(int empno, double newSal)
手書き/有効	throws SQLException {
手書き/有効	System.out.println("static update-returning for empno " + empno);

```
手書き/有効                    String ename;
手書き/SQL文                   #sql { begin
手書き/SQL文                        update emp set sal = :newSal
手書き/SQL文                        where empno = :empno
手書き/SQL文                        returning ename into :OUT ename; -- :OUT is for SQLJ bind
手書き/コメント                                        -- :OUT is for SQLJ bind
手書き/SQL文                        end;
手書き/SQL文                   };
手書き/有効                    System.out.println("Update the salary of employee " + ename);
手書き/有効               }
組込み/コメント      //ユーザ領域終了>>
組込み/有効      }
              [EOF]
```

HTML/JSP/JS/CSS計測基準

COBOL計測基準	C/C++計測基準	Java計測基準	IDL計測基準	Java自動生成ソース計測基準	HTML計測基準	.NET計測基準	VB6.0計測基準

HTML/JSP/JS/CSSの計測基準は以下のようになります。

注記判定

・ HTML、JSPファイルの「<!--」と「-->」で括られたステップ。（注１、注２）
・ HTML、JSPファイルの「<comment>」と「</comment>」で括られたステップ。（注２）
・ JSPファイルの「<%--」と「--%>」で括られたステップ。（注２）
・ JSPファイルの「<%」と「%>」で括られた範囲内にある「//」で開始されるステップ。
・ JSPファイルの「<%」と「%>」で括られた範囲内にある「/*」と「*/」で括られるステップ。
・ HTML、JSPファイルの<SCRIPT>タグで括られたステップ。（注３）
・ HTML、JSPファイルの<STYLE>タグで括られたステップ。
・ JSまたはCSSファイル内の「//」で開始されるステップ。
・ JSまたはCSSファイル内の「/*」と「*/」で括られたステップ。
・ 空行。

注１） <SCRIPT>タグで「language」及び「type」の属性値としてJavaScript、VBScriptが指定された場合、「<!--」と「-->」で括られた記述は各スクリプト／言語の注釈判定により計測されます。上記以外のスクリプトが指定された場合、計測エラーとなり<SCRIPT>～</SCRIPT>範囲はコメント行として計測されます。
注２） <XMP>タグで括られたコメント表記や、<PLAINTEXT>タグ以降に記述したコメント表記は対象外となり、有効行と計測されます。
注３） <SCRIPT>タグで「language」及び「type」の属性値が省略されている場合、<SCRIPT>タグで括られた記述はJavaScriptの注釈判定により計測されます。

備考

・ 「language」及び「type」属性が記述される<SCRIPT>タグは１行で記述されていなければ正確な計測ができません。
　<SCRIPT>タグの途中で改行されている場合、前回指定された<SCRIPT>タグのスクリプト属性で計測されます。

計測基準例

判定	ソースプログラム記述(sample.html)

```
有効        <HTML>
有効         <HEAD>
コメント     <!-- ステップ数計測機能　HTML ソース計測の例 -->
コメント     <!-- ALL Rights Reserved Copyright FUJITSU LIMITED 2004 -->
コメント
有効         </HEAD>
有効         <BODY>
コメント       <COMMENT>テスト資産です</COMMENT>
有効           <SCRIPT LANGUAGE="JavaScript">
有効           <!--
コメント       //test01関数
有効           function test01(){
コメント       /*そのまま復帰*/
有効           return true;
有効           }
有効           -->
有効           </SCRIPT>
有効           <SCRIPT LANGUAGE="JavaScript">
有効           <!--
コメント       //test01関数
有効           function test01(){
コメント       /*そのまま復帰*/
有効           return true;
有効           }
有効           -->
有効           </SCRIPT>
有効           <STYLE TYPE="text/css">
有効           <!--
コメント       /*スタイル*/
有効           name {background-color:#ffcccc}
有効           -->
有効           </STYLE>
有効         </BODY>
有効        </HTML>
            [EOF]
```

.NET(ASP/C#/VB)/HTML/JS/CSS計測基準

ASP.NET、C#、VB.NET資産、及びHTML、JS、CSSを含めた計測が可能です。
(以下、ASP/ASP.NET を ASP、Visual Basic/Visual Basic.NET を VB と略します。)
.NET(ASP/C#/VB)/HTML/JS/CSSの計測基準は以下のようになります。

注記判定

・ ASP、HTMLファイルの「<!--」と「-->」で括られたステップ。（注１、注２）
・ ASP、HTMLファイルの「<comment>」と「</comment>」で括られたステップ。（注２）
・ ASPファイルの「<%--」と「--%>」で括られたステップ。（注２）
・ ASPファイルの「<%」と「%>」で括られたステップ。（注３）
・ ASP、HTMLファイルの<SCRIPT>タグで括られたステップ。（注４）
・ ASP、HTMLファイルの<STYLE>タグで括られたステップ。
・ C#、JS、CSSファイル内の「//」で開始されるステップ。
・ C#、JS、CSSファイル内の「/*」と「*/」で括られたステップ。
・ VB、VBScriptファイル内の「Rem」で開始されるステップ。
・ VB、VBScriptファイル内の「'」で開始されるステップ。
・ 空行。

注１）<SCRIPT>タグで「language」及び「type」の属性値としてC#、VB、JavaScript、VBScript、JScriptが指定された場合、「<!--」と「-->」で括られた記述は各スクリプト／言語の注釈判定により計測されます。上記以外のスクリプトが指定された場合、計測エラーとなり<SCRIPT>～</SCRIPT>範囲はコメント行として計測されます。
注２）<SCRIPT>タグ、<STYLE>タグ、<XMP>タグ、「<%@」と「 %>」、「<%」と「%>」で括られたコメント表記や、<PLAINTEXT>タグ以降に記述したコメント表記は対象外となり、有効行と計測されます。
注３）@languageディレクティブでC#、VB、JavaScript、VBScript、JScriptが指定された場合、「<%」と「%>」で括られた記述は各スクリプト／言語の注釈判定により計測されます。@languageディレクティブが省略されている場合、「<%」と「%>」で括られた記述はVBScriptの注釈判定により計測されます。
注４）<SCRIPT>タグで「language」及び「type」の属性値が省略されている場合、<SCRIPT>タグで括られた記述は、@languageディレクティブで指定されたスクリプト属性により計測されます。

備考

・ 「language」及び「type」属性が記述される<SCRIPT>タグは１行で記述されていなければ正確な計測ができません。
<SCRIPT>タグの途中で改行されている場合、以下のスクリプト属性で計測されます。

－ASPファイルの場合

<SCRIPT>タグの改行位置によって、@languageディレクティブで指定されたスクリプト属性または前回指定された<SCRIPT>タグのスクリプト属性で計測されます。

－HTMLファイルの場合

前回指定された<SCRIPT>タグのスクリプト属性で計測されます。

・<SCRIPT>タグや@languageディレクティブに指定可能なスクリプト属性は、計測対象資産の拡張子によって異なります。

拡張子	対象スクリプト
*.asp	JavaScript、VBScript、JScript
*.aspx	C#、VB、JavaScript、VBScript、JScript
.htm、.html	JavaScript、VBScript、JScript

計測基準例

判定	ソースプログラム記述(sample.aspx)
有効	<%@ Page ContentType="text/html" Language="VB" %>
コメント	<%--
コメント	＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊
コメント	ステップ数計測機能　ASP.NET ソース計測の例
コメント	ALL Rights Reserved Copyright FUJITSU LIMITED 2004
コメント	＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊
コメント	--%>
コメント	
有効	<HTML>
有効	<HEAD>
有効	<TITLE>ASP.NET Sample</TITLE>

```
コメント	<%-- サンプルソース --%>
有効	<meta content="JavaScript" name="vs_defaultClientScript">
コメント
有効	<script runat="Server">
有効	Sub Page_Load(sender As Object, e As EventArgs)
コメント	    'ページの初期化
有効	    TextBox.text = "テキストボックス"
有効	    Button01.Attributes.add("OnClick","alert('ボタン１が押下されました。');")
有効	End Sub
有効	</script>
有効	<script language="JavaScript">
有効	<!--
コメント	    /* test01関数 */
有効	    function test01(){
コメント	        //メッセージを表示します
有効	        window.alert("ボタン２が押下されました。");
有効	    }
有効	-->
有効	</script>
コメント
有効	</HEAD>
有効	<body>
有効	    <form runat="server">
コメント	        <!-- コメント -->
有効	        <h1>サンプルソース</h1>
有効	        <P><asp:textbox id="TextBox" runat="server"></asp:textbox></P>
有効	        <asp:button id="Button01" runat="server" Text="ボタン１"></asp:button><br>
有効	        <br>
有効	        <input type="button" OnClick="test01();" value="ボタン２">
有効	    </form>
有効	</body>
有効	</HTML>
```

```
        [EOF]

判定    ソースプログラム記述(sample.cs)
コメント  /* ステップ数計測機能　C# ソース計測の例 */
コメント  /* ALL Rights Reserved Copyright FUJITSU LIMITED 2004 */
有効    using System;
コメント
有効    class HelloClass
有効    {
有効        static void Main()
有効        {
コメント          // Hello worldを表示します。
有効            Console.WriteLine("Hello World!");
有効        }
有効    }
        [EOF]

判定    ソースプログラム記述(sample.vb)
コメント  '* * * * * * * * * * * * * * * * * * * * * * * * * * *
コメント  ' ステップ数計測機能　VisualBasic.NET ソース計測の例
コメント  ' ALL Rights Reserved Copyright FUJITSU LIMITED 2004
コメント  '* * * * * * * * * * * * * * * * * * * * * * * * * * *
有効    Imports System
有効    Class Hellow
コメント
有効        Shared Sub Main()
コメント          Rem Hello worldを表示します。
有効            Console.WriteLine("Hello world!")
有効        End Sub
コメント
有効    End Class
```

```
        [EOF]

判定    ソースプログラム記述(sample.asp)
有効    <%@ LANGUAGE="VBScript" %>
有効    <%
コメント '***************************
コメント ' ステップ数計測機能　ASP ソース計測の例
コメント ' ALL Rights Reserved Copyright FUJITSU LIMITED 2004
コメント '***************************
有効    %>
コメント
有効    <HTML>
有効    <HEAD>
コメント   <%-- サンプルソース --%>
有効      <META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
有効      <META http-equiv="Content-Script-Type" content="text/javascript">
有効      <META http-equiv="Content-Style-Type" content="text/css">
有効      <META http-equiv="Pragma" content="no-cache">
有効      <LINK rel="StyleSheet" href="sample.css">
有効      <TITLE>ASP Sample</TITLE>
コメント
有効    <SCRIPT LANGUAGE="JavaScript">
有効    <!--
コメント   /* test01関数 */
有効      function test01(){
コメント     //メッセージを表示します
有効        window.alert("ボタンが押下されました。");
有効        return true;
有効      }
有効    -->
有効    </SCRIPT>
有効    </HEAD>
```

```
コメント
有効	<BODY>
有効	<%
コメント	  '文字列を表示します
有効	  Response.Write "Hello World!!"
有効	%>
有効	  <DIV align="center">
有効	  <TABLE>
有効	    <INPUT type="submit" onClick="return test01();" name="Move" value="　テスト　">
有効	  </TABLE>
有効	  </DIV>
有効	</BODY>
有効	</HTML>
	[EOF]
```

COBOL計測基準	C/C++計測基準	Java計測基準	IDL計測基準	Java自動生成ソース計測基準	HTML計測基準	.NET計測基準	VB6.0計測基準

Visual Basic 6.0の計測基準は以下のようになります。

注記判定

- 「Rem」で開始されるステップ。
- 「'」で開始されるステップ。
- 空行。

手書き・自動生成部分の判定

・手書き・自動生成部分は、自動識別子オプションダイアログで設定する、開始識別子・終了識別子をもとに判定され、開始識別子の行から終了識別子の行までを自動生成部分とみなし、それ以外の部分を手書き部分とみなし計測します。
Java自動生成ソースの計測基準とは逆になりますので、ご注意ください。
・自動識別子は前方一致で検索されます。
・以下の識別子は、判定の対象外となるため、注意が必要です。
　－１つの開始識別子・終了識別子の範囲内に存在する、入れ子状態の開始識別子・終了識別子
　－開始識別子・終了識別子の組み合わせが不当である場合の終了識別子
　－開始識別子が存在しない終了識別子
・自動識別子が1対も無いソースの計測結果は、全て手書きソース（手書きステップ数）として計測されます。

備考

- 計測結果出力時は、自動生成部分を組込みステップ情報として出力します。
- 標準で登録されている識別子については、VB6.0自動識別オプションの設定を参照ください。

計測基準例

判定	ソースプログラム記述(sample.frm)
手書き/コメント	'**************************
手書き/コメント	' ステップ数計測機能　VisualBasic ソース計測の例
手書き/コメント	' ALL Rights Reserved Copyright FUJITSU LIMITED 2004
手書き/コメント	'**************************
手書き/コメント	
組込み/有効	VERSION 5.00
組込み/コメント	
組込み/有効	Begin VB.Form Form1
組込み/有効	Caption = "Form1"
組込み/有効	ClientHeight = 3195
組込み/有効	ClientLeft = 60
組込み/有効	ClientTop = 345
組込み/有効	ClientWidth = 4680
組込み/有効	LinkTopic = "Form1"
組込み/有効	ScaleHeight = 3195
組込み/有効	ScaleWidth = 4680
組込み/有効	StartUpPosition = 3 'Windows の既定値
組込み/有効	End
組込み/コメント	
組込み/有効	Attribute VB_Name = "Form1"
組込み/有効	Attribute VB_GlobalNameSpace = False
組込み/有効	Attribute VB_Creatable = False
組込み/有効	Attribute VB_PredeclaredId = True
組込み/有効	Attribute VB_Exposed = False
手書き/コメント	
手書き/有効	Imports System
手書き/有効	Class Hellow
手書き/コメント	
手書き/有効	Shared Sub Main()
手書き/コメント	Rem Hello worldを表示します。
手書き/有効	Console.WriteLine("Hello world!")
手書き/有効	End Sub

手書き/コメント
手書き/有効　End Class
[EOF]

制限事項

注意事項

SIMPLIA/MF-STEPCOUNTERを使用するにあたっての制限事項を以下に示します。

・ SIMPLIA/MF-STEPCOUNTERでは、260バイト以上のパスを取り扱うことができません。

・ 4GByte以上のソースファイルを計測することはできません。

・ サーチパスに指定できるパス数は10件までです。11件以上のパスの指定はできません。

・ 組込みメンバを計測対象ファイルとして計測した場合、正常な計測ができません。

・ 組込みメンバの入れ子の階層は8階層までです。それ以上は計測を行いません。

・ 計測対象資産は、各言語の文法に従って正しく記述されていることを前提としています。

・ 印刷用紙の大きさはA4縦の固定になります。それ以外の様式を指定した場合、印刷結果は保証されません。

・ 帳票の注釈欄の文字数は日本語で15文字、英数半角文字で30バイトまでです。それ以上の文字数は指定できません。

・ 埋め込みSQLのインクルード（EXEC SQL INCLUDE,$include)で指定されたインクルードファイルは組込みメンバの対象とはなりません。

・ 埋め込みSQL中の登録集やインクルード(#include等）は組込みメンバの対象にはなりません。

・ 平均や率の計算は、全て小数点第一位で四捨五入しています。

・ 平均行の小数点以下の丸め誤差により、特に計測対象資産数や資産規模などが小さい場合に、値の矛盾や異常な値（100%以上など）が表示されることがあります。

・ 帳票形式とリストビュー形式で平均行の各値の算出方法が異なるため、値が異なる場合があります。

帳票形式について

・ ファイル名の出力可能な長さは言語種別や帳票オプション（SQLステップ数の出力、ファイル日付の出力）により変化します。各状況による最大長以後のファイル名は出力されません。

リストビュー形式について

・ 注釈率下限値の判定について

比較対象の値は表示用データです。表示データは小数点第一位で四捨五入されているため、本来対象とならないものまで赤色表示の対象となります。

例) 注釈率下限値:20%以下を指定している場合、以下のケースでも赤色表示される。

20% （本当は20.1%だが四捨五入しているため、画面上20%で表示されている）

・ Java言語時のファイル種別の判定について

１つのファイル中に複数のクラス／インタフェース定義が存在する場合は、最初のクラス／インタフェースで判定します。

・ リストビュー形式の印刷はできません。CSV形式ファイルへ出力後、Excelなどの表計算ソフトなどで編集して印刷してください。

・ 各列に対するソートの昇順／降順指定は、リストビュー形式画面を閉じると初期化されます。また、プログラム情報と組込みメンバ情報の両方が表示されている場合、それぞれの同項目間での同期はとりません。

・ 表示フォルダの変更は、リストビュー形式画面を閉じると初期化されます。

・ 注釈率は「１００％－［宣言部有効ステップ数％］－［実行部有効ステップ数％］」で計算されるため、注釈ステップ数と合計ステップ数から計算した注釈率と合わない場合があります。

計測対象資産について

・ 計測対象資産中に300バイトを超えるレコードが存在する場合、計測結果は保証されません。

・ 計測対象資産中に「CR+LF」以外の改行コードが存在する場合、計測結果は保証されません。

・ ShiftJISまたは、UNICODE（UTF-8およびBOM(Byte Order Mark)ありのUCS2（ビッグエンディアンまたはリトルエンディアン））で記述されるソースファイルを対象としているため、これ以外の文字コードで記述されているソースファイルの計測結果は保証されません。

COBOLの計測について

・ ソース中に見出し部の開始（IDENTIFICATION DIVISION）と手続き部の開始（PROCEDURE DIVISION）が記述されていなければなりません。一方あるいは両方が未記述やコピー句登録集に記述されている場合、正常な計測ができません。

・ COBOLの自由形式の正書法には対応しておりません。

・ プログラムやコピー句登録集の一連番号領域(1～6カラム目)、標識領域(7カラム目)、A領域(8～12カラム目)に跨ってタブ表記が使用される場合、正常に計測されません。

・ 組込みメンバを計測する場合でフォルダ単位で計測対象資産を指定したとき、組込みメンバが指定フォルダと同一にある場合や、指定フォルダのサブフォルダにあり、計測オプションでサブフォルダも検索する指定になっている場合、それらのフォルダをサーチパスに指定していても、組込みメンバもプログラムステップとして計測されてしまいます。組込みメンバは指定フォルダ配下以外に配置するようにしてください。

C/C++の計測について

・ 文字列定数の継続行に記述されている、#include,rcincludeは組込み命令として認識します。

例) 以下の場合、「AAA.H¥」を組込みメンバ名と見なし計測対象としてサーチします。

```
strcpy(szBuf,"ABCDEFG ¥
```

```
#include ¥"AAA.H¥"");
```

Java自動生成ソース・Visual Basic 6.0の計測について

・ 100件を超える識別子の登録はできません。

・ Java自動生成ソース及び、Visual Basic 6.0の計測において以下の識別子は、判定対象外です。
　－１つの開始識別子・終了識別子の範囲内に含まれる、入れ子状態の開始識別子と終了識別子
　－開始識別子と組み合わせの合わない不当な終了識別子
　－開始識別子の存在しない終了識別子

・ 標準で登録されているVB6.0用の識別子はマイクロソフト社の公開された仕様に基づいたものではありません。このため、正確に計測されない場合は識別子を編集／追加してください。

・ 標準で提供されているVB6.0自動識別オプションでは以下の制限があります。
　－*.clsファイルによっては"Attribute VB_Exposed"属性の記述行以降にも，"Attribute VB_Ext_KEY"という属性を記述した行があるものが確認されています。このような行は手書き行として計測されてしまいます。

ASPの計測について

・ 「language」及び「type」属性が記述される<SCRIPT>タグが、途中で改行されている場合、誤ったスクリプト属性で計測されてしまいます。

文字コードの指定について

・誤った文字コードが指定されている場合、計測結果は保証されません。

・異なる文字コードが使用されている計測対象資産を同時に計測した場合、指定した文字コード以外のファイルの計測結果は保証されません。

注意事項

制限事項

SIMPLIA/MF-STEPCOUNTERを使用するにあたっての注意事項を以下に示します。

・ SIMPLIA/MF-STEPCOUNTERが使用するフォントとして、以下のフォントがWindowsのシステムに組み込まれている必要があります。

－ MS 明朝

・ ディスプレイ解像度は、640×480以上を使用してください。

・ 組込みメンバのフォルダの検索ついて

サーチパスオプションで指定したフォルダとは別に対象資産と同一フォルダ内も検索します。サーチパスオプションに指定が無い場合は、対象資産と同一フォルダ内のみ検索します。

優先順位は、サーチパスオプションで指定したフォルダ内を先に検索します。

・ 計測オプション「組込みメンバを計測する」をチェックした場合の計測処理時間は、組込みメンバの呼び出し数や構成・階層の深さといった要素に比例し、複雑なほど多くの時間を要します。必要でない限り、チェックをはずしてご使用されることをお勧めします。

・ 計測の言語種別に「Java自動生成ソース」を選択した場合、Java自動識別オプションの設定で指定した識別子が1対も無いソースの計測結果は、全て自動生成ソース（組込みステップ数）として計測されます。

・ 計測の言語種別に「Visual Basic 6.0」を選択した場合、VB6.0自動識別オプションの設定で指定した識別子が1対も無いソースの計測結果は、全て手書きステップ数として計測されます。

エラーメッセージ

- 中間ファイル[ファイル名]のオープンに失敗しました。
- 中間ファイル[ファイル名]の書き込みに失敗しました。
- 計測結果の出力に失敗しました。
- 計測結果ファイル[ファイル名]のオープンに失敗しました。
- 計測結果ファイル[ファイル名]の書き込みに失敗しました。
- 計測結果ファイル[ファイル名]の形式が正しくありません。
- 計測対象ファイル[ファイル名]の形式が正しくありません。
- 計測対象ファイル[ファイル名]のオープンに失敗しました。
- 計測対象ファイル[ファイル名]の注釈表記([数値]行目開始)が正しくありません。
- 計測対象ファイル[ファイル名]のＳＱＬ表記([数値]行目開始)が正しくありません。
- 計測エラーが[数値]件ありました。エラーファイルを確認して下さい。
- 大きすぎて出力できないステップ数がありました。
- ＣＳＶ形式ファイル[ファイル名]のオープンに失敗しました。
- ＣＳＶ形式ファイル[ファイル名]の書き込みに失敗しました。
- 環境変数TEMPの値が正しくありません。[ディレクトリ名]を作業用ディレクトリとします。
- メモリ不足のため、リストボックスへの文字列追加ができません。
- サーチパス件数が多すぎます。最大[数値]件まで指定可能です。
- 対象が多すぎます。一度に選択できるのは[数値]件までです。操作を何回かに分けて下さい。
- ページ指定に誤りがあります。
- 部数指定に誤りがあります。
- プリンタが使用できません。
- スプール用のディスク容量が不足しているため印刷できません。

スプール用のメモリが不足しているため印刷できません。

ドキュメント種別の指定に誤りがあります。

計測結果が保存されていません。保存しますか？

メモリ不足のためメッセージ通知ができませんでした。

作業用ディレクトリ[ディレクトリ名]の作成に失敗しました。SIMPLIA/MF-STEPCOUNTERを起動できません。

TEMPフォルダ名の取得に失敗しました。

対象資産フォルダ名ファイルのオープンに失敗しました。

対象資産ファイル名ファイルのオープンに失敗しました。

指定されたフォルダは存在しません。

指定された文字列はフォルダ名ではありません。

同名のフォルダが既に登録されています。

指定されたフォルダの下位階層フォルダが既に登録されています。

指定されたフォルダの上位階層フォルダが既に登録されています。

サポートしていない形式の計測結果ファイルです。

組込みメンバステップ情報の計測結果がありませんでした。

識別文字列は、最大件数件以上指定できません。

識別文字列の指定が重複しています。

識別子の組み合わせが不当です。

識別子ファイルの読み込みに失敗しました。

識別子ファイルの書き出しに失敗しました。

設定内容を消去し、外部ファイルの識別子を取り込みます。

計測対象ファイル[ファイル名]のタグ表記([数値]行目開始)が正しくありません。

このオプションを有効にすると、COBOLソースの計測がプログラムID単位となります。

また、計測結果のファイル名にプログラムID情報が付加されて表示されます。
副プログラムを計測したくない場合は、このオプションを無効にしてください。

このオプションを無効にすると、COBOLソースの計測はメインプログラムのみとなります。
また、計測結果のファイル名にプログラムID情報は付加されません。
副プログラムも含めて計測したい場合は、このオプションを有効にしてください。

このオプション設定時は、「フォルダ指定の場合、ヘッダーファイル(*.h)も計測する」機能は使用できません。

古いバージョンの計測結果ファイルが読み込まれました。
プログラムID単位に計測する場合は、計測オプションの設定を見直してください。

古いバージョンの計測結果ファイルが読み込まれました。
ヘッダーファイル(*.h)をプログラムステップ情報として計測する場合は、計測オプションの設定を見直してください。

中間ファイル[ファイル名]のオープンに失敗しました。

MF-STEPCOUNTERが内部で使用する中間ファイルのオープンに失敗しました。中間ファイルは[環境変数TEMPに設定されているディレクトリ¥MFSTPTMP]という名前のディレクトリ配下にまとめて格納され、MF-STEPCOUNTER終了時にMFSTPTMPのディレクトリごと削除されます。

想定される原因

1. MF-STEPCOUNTER起動中に中間ファイルが削除された。
2. 媒体不良。

対処方法

環境変数TEMPに設定されているディレクトリ(ドライブ)が使用可能かどうか確認してください。

中間ファイル[ファイル名]の書き込みに失敗しました。

MF-STEPCOUNTERが内部で使用する中間ファイルの書き込みに失敗しました。中間ファイルは[環境変数TEMPに設定されているディレクトリ¥MFSTPTMP]という名前のディレクトリ配下にまとめて格納され、MF-STEPCOUNTER終了時にMFSTPTMPのディレクトリごと削除されます。

想定される原因

1. 環境変数TEMPに設定されているディレクトリ(ドライブ)に空きがない。
2. 媒体不良。

対処方法

環境変数TEMPに設定されているディレクトリ(ドライブ)が使用可能かどうか確認してください。

計測結果の出力に失敗しました。

画面またはプリンタへのドキュメント出力処理で異常が発生しました。

想定される原因

1. 中間ファイルが正常に作成されなかった。(中間ファイルのオープン、書き込みエラーがその前に出ていた。）
2. 読み込みの場合、指定された計測結果ファイルの内容が正しくない。

対処方法

読み込みの場合は正しい内容の計測結果ファイルを指定してください。

計測結果ファイル[ファイル名]のオープンに失敗しました。

指定された計測結果ファイルのオープンに失敗しました。

想定される原因

1. 読み込みの場合、指定された計測結果ファイルが存在しない。(名前が正しくない。)
2. 媒体不良。

対処方法

計測結果ファイルの格納先ドライブが使用可能かどうか確認してください。読み込みの場合は正しい計測結果ファイル名を指定してください。

計測結果ファイル[ファイル名]の書き込みに失敗しました。

指定された計測結果ファイルの書き込みに失敗しました。

想定される原因

1. 計測結果ファイル格納先ディレクトリ(ドライブ)に空きがない。
2. 媒体不良。

対処方法

計測結果ファイルの格納先ドライブが使用可能かどうか確認してください。

計測結果ファイル[ファイル名]の形式が正しくありません。

指定された計測結果ファイルの内容に誤りがあります。

想定される原因

1. 計測結果ファイル以外のファイルを指定して読み込みを行った。
2. 計測結果ファイルの内容が破壊されている。
3. 媒体不良。

対処方法

正しい内容の計測結果ファイルを指定してください。

計測対象ファイル[ファイル名]の形式が正しくありません。

計測対象のソースファイルの内容に誤りがあります。(メッセージの出力先はディスプレイ、エラーファイルの両方です。)

想定される原因

1. 計測対象としてテキストファイル以外のファイルを指定して計測を行った。

対処方法

計測対象にはテキストファイル以外を指定することができません。正しいソースファイルを計測対象として指定してください。

計測対象ファイル[ファイル名]のオープンに失敗しました。

計測対象ソースファイルのオープンに失敗しました。

想定される原因

1. 指定されたソースファイルが存在しない。
2. 媒体不良。

対処方法

現在、計測対象ソースファイルが存在するかどうか確認してください。

計測対象ファイル[ファイル名]の注釈表記([数値]行目開始)が正しくありません。

計測対象ソースファイル内の注釈表記に誤りがあります。(C／C++、Java、Java自動生成ソース、IDLの計測時に判定、通知されます。またメッセージの出力先はディスプレイ、エラーファイルの両方です。)

想定される原因

1. 注釈開始記号(/*)またはJavadocコメント開始記号(/**)に対応する注釈終了記号(*/)が発見されない状態で計測対象ソースファイルの終端に達した。

対処方法

計測対象ソースファイル内の注釈表記を確認してください。

計測対象ファイル[ファイル名]のＳＱＬ表記([数値]行目開始)が正しくありません。

　計測対象ソースファイル内の埋め込みSQL表記に誤りがあります。(計測オプションでSQLありが指定された場合のみ通知されます。またメッセージの出力先はディスプレイ、エラーファイルの両方です。）

想定される原因

1. 埋め込みSQL開始記号($,EXEC SQL等)に対応する埋め込みSQL終了記号(;,END-EXEC等)が発見されない状態で計測対象ソースファイルの終端に達した。

対処方法

計測対象ソースファイル内の埋め込みSQL表記を確認してください。

計測エラーが[数値]件ありました。エラーファイルを確認して下さい。

計測対象ソースファイル内に注釈表記エラーまたは埋め込みSQL表記エラーがありました。エラーファイル(環境変数TEMPに設定されているディレクトリ¥MF_STP.ERR)を確認してください。

想定される原因

1. 計測対象ソースファイル内に注釈表記エラー、埋め込みSQL表記エラー等があった。

対処方法

エラーファイルを確認してエラーが存在するソースファイルを修正してください。

大きすぎて出力できないステップ数がありました。

計測対象ソースファイルのステップ数がMF-STEPCOUNTERの許容範囲を超えました。計測結果として、ステップ数は"**********"、パーセントは"***"と出力されます。

想定される原因

1. ステップ数(合計、のべ総ステップも含む)が99,999,999を超えた。

対処方法

99,999,999ステップ以上のドキュメント出力はできません。

ＣＳＶ形式ファイル[ファイル名]のオープンに失敗しました。

CSV形式ファイル作成時に、指定されたファイルをオープンできませんでした。

想定される原因

1. 媒体不良。(格納先がフロッピーディスクの場合、フロッピーディスクが差し込まれていない。)

対処方法

CSV形式ファイルを作成するディレクトリ(ドライブ)が使用可能かどうか確認してください。

ＣＳＶ形式ファイル[ファイル名]の書き込みに失敗しました。

CSV形式ファイル作成時に、指定されたファイルをオープンできませんでした。

想定される原因

1. CSV形式ファイルを作成するディレクトリ(ドライブ)に空きがない。
2. 媒体不良。

対処方法

CSV形式ファイルを作成するディレクトリ(ドライブ)が使用可能かどうか確認してください。

環境変数TEMPの値が正しくありません。[ディレクトリ名]を作業用ディレクトリとします。

環境変数TEMPに設定されているドライブ、ディレクトリが無効です。[ディレクトリ名]をMF-STEPCOUNTERの作業用ディレクトリとし、中間ファイルを[ディレクトリ名]¥MFSTPTMPという名前のディレクトリ配下に作成します。なおエラーファイルは［ディレクトリ名］配下に作成します。

想定される原因

1. 環境変数TEMPが未定義である。
2. 環境変数TEMPに設定されているディレクトリが存在しない。
3. 媒体不良。(フロッピーディスクの場合差し込まれていない。)

対処方法

環境変数TEMPに有効なパスを設定してください。([ディレクトリ名]に十分な空きがあればこのままでもMF-STEPCOUNTERは動作しますが、環境変数TEMPに設定されてるディレクトリ(ドライブ)はWindowsが作業領域として使用しますので正しい値を設定してください。)

メモリ不足のため、リストボックスへの文字列追加ができません。

新規計測ダイアログボックスまたは対象ファイル編集ダイアログボックスで、リストボックスへの文字列追加の際メモリ不足が発生しました。

想定される原因

1. ファイル、ディレクトリ、対象ファイル一覧リストボックスのいずれかにメモリの許容範囲を超える件数の項目を追加(表示)しようとした。

対処方法

リストボックスへ追加可能な項目の最大件数はメモリ容量によって制限されます。それを超える件数の項目追加はできません。

サーチパス件数が多すぎます。最大[数値]件まで指定可能です。

サーチパスの指定件数がMF-STEPCOUNTERの制限値を超えました。

想定される原因

1. ファイル、ディレクトリ、対象ファイル一覧リストボックスのいずれかにメモリの許容範囲を超える件数の項目を追加(表示)しようとした。

対処方法

件数の制限値内でサーチパスの指定を行ってください。

対象が多すぎます。一度に選択できるのは[数値]件までです。操作を何回かに分けて下さい。

新規計測ダイアログボックスまたは対象ファイル編集ダイアログボックスで対象ファイル一覧への追加、削除操作を行う際の選択件数(反転表示項目数)がMF-STEPCOUNTERの制限値を超えました。

想定される原因

1. 選択件数(反転表示項目数)が制限値を超えた。

対処方法

追加、削除操作を何回かに分割して一回の選択件数を制限値内に収めてください。

ページ指定に誤りがあります。

ウィンドウメニューのページ指定ダイアログボックスまたは印刷範囲指定ダイアログボックスで誤ったページが指定されました。

想定される原因

1. 計測結果として作成されたドキュメントとして存在しないページ数が指定された。
2. 数値以外の文字(空白、タブを含む)が指定された。

対処方法

正しいページ数を指定してください。

部数指定に誤りがあります。

印刷範囲指定ダイアログボックスで誤った部数が指定されました。

想定される原因

1. 0以下の数値を指定した。
2. 数値以外の文字(空白、タブを含む)が指定された。

対処方法

正しい部数を指定してください。

プリンタが使用できません。

印刷に使用するプリンタが使用可能状態にありません。

想定される原因

1. 通常使うプリンタが設定されていない。

対処方法

コントロールパネルを使用して通常使うプリンタを設定してください。

スプール用のディスク容量が不足しているため印刷できません。

印刷時に使用するスプールディレクトリ(環境変数TEMPに設定されているディレクトリ)の容量不足のため印刷を行うことができません。

想定される原因

1. スプールディレクトリの容量不足。

対処方法

環境変数TEMPに十分な空きがあるディレクトリ(ドライブ)を指定してください。

スプール用のメモリが不足しているため印刷できません。

印刷時に使用するスプールメモリの容量不足のため印刷を行うことができません。

想定される原因

1. メモリの容量不足。

対処方法

メモリの制限を超える印刷できません。

ドキュメント種別の指定に誤りがあります。

計測オプション設定ダイアログボックスまたは印刷ダイアログボックスでのドキュメント種別指定に誤りがあります。

想定される原因

1. 出力、計測するドキュメントとして何も指定(チェック)されなかった。

対処方法

出力、計測対象として最低一つのドキュメントを指定しなければなりません。ダイアログボックス内での設定値を取り消す場合は「キャンセル」を選択してください。

計測結果が保存されていません。保存しますか？

現在の計測結果が破棄されようとしています。よろしいですか?

想定される原因

未保存の計測結果が画面上にある状態で以下に示すいずれかの操作が行われた。

1. 「新規計測」が選択された。
2. ファイルの読み込みダイアログボックスで計測結果ファイルを指定してOKボタンが押された。
3. MF-STEPCOUNTERの終了メニューが選択された。
4. Windowsを終了させた。

対処方法

保存する場合は「はい」、破棄する場合は「いいえ」を選択してください。

メモリ不足のためメッセージ通知ができませんでした。

メモリ不足のためメッセージ通知用のメッセージボックスを生成することができませんでした。(このメッセージはエラーファイルに出力されます。)

想定される原因

1. メモリ不足

対処方法

他に起動中のアプリケーションがあればそれらを終了してください。

作業用ディレクトリ[ディレクトリ名]の作成に失敗しました。SIMPLIA/MF-STEPCOUNTERを起動できません。

中間ファイル格納用一時ディレクトリを作成することができませんでした。

想定される原因

1. 環境変数TEMPに設定されているディレクトリ(ドライブ)に空きがない。
2. 媒体不良

対処方法

環境変数TEMPに設定されているディレクトリ(ドライブ)が使用可能かどうか確認してください。

TEMPフォルダ名の取得に失敗しました。

環境変数TEMPに設定されているディレクトリ名に取得に失敗しました。

想定される原因

1. OSの不良。

対処方法

環境変数TEMPに設定されているディレクトリ(ドライブ)が使用可能かどうか確認してください。

対象資産フォルダ名ファイルのオープンに失敗しました。

MF-STEPCOUNTERが内部で使用する中間ファイル[対象資産フォルダ名ファイル]のオープンに失敗しました。中間ファイルは[環境変数TEMPに設定されているディレクトリ¥MFSTPTMP]という名前のディレクトリ配下にまとめて格納され、MF-STEPCOUNTER終了時にMFSTPTMPのディレクトリごと削除されます。

想定される原因

1. MF-STEPCOUNTER起動中に中間ファイルが削除された。
2. 媒体不良。

対処方法

環境変数TEMPに設定されているディレクトリ(ドライブ)が使用可能かどうか確認してください。

対象資産ファイル名ファイルのオープンに失敗しました。

MF-STEPCOUNTERが内部で使用する中間ファイル[対象資産ファイル名ファイル]のオープンに失敗しました。中間ファイルは[環境変数TEMPに設定されているディレクトリ¥MFSTPTMP]という名前のディレクトリ配下にまとめて格納され、MF-STEPCOUNTER終了時にMFSTPTMPのディレクトリごと削除されます。

想定される原因

1. MF-STEPCOUNTER起動中に中間ファイルが削除された。
2. 媒体不良。

対処方法

環境変数TEMPに設定されているディレクトリ(ドライブ)が使用可能かどうか確認してください。

指定されたフォルダは存在しません。

指定されたフォルダは存在しません。

想定される原因

1. 指定された名前のフォルダが存在しない。

対処方法

フォルダが存在するか確認してください。

指定された文字列はフォルダ名ではありません。

指定された文字列はフォルダ名ではありません。

想定される原因

1. ファイル名である。

対処方法

フォルダを確認してください。

同名のフォルダが既に登録されています。

同名のフォルダが既に登録されています。

想定される原因

1. 同名のフォルダが既に登録されている。

対処方法

同一フォルダを重複して登録することはできません。

指定されたフォルダの下位階層フォルダが既に登録されています。

指定されたフォルダの下位階層フォルダが既に登録されています。

想定される原因

1. 計測オプションの[サブフォルダを検索する]がチェックされている。

対処方法

下位階層フォルダを削除するか、計測オプションの[サブフォルダを検索する]を外してください。

指定されたフォルダの上位階層フォルダが既に登録されています。

指定されたフォルダの上位階層フォルダが既に登録されています。

想定される原因

1. 計測オプションの[サブフォルダを検索する]がチェックされている。

対処方法

上位階層フォルダを削除するか、計測オプションの[サブフォルダを検索する]を外してください。

サポートしていない形式の計測結果ファイルです。

サポートしている計測結果ファイルの形式ではありませんでした。

想定される原因

1. サポートしていないバージョンのMF-STEPCOUNTERで作成された計測結果ファイルである。
2. 計測結果ファイルでは無い。

対処方法

指定したファイルを確認してください。

組込みメンバステップ情報の計測結果がありませんでした。

組込みメンバステップ情報の計測結果がありませんでした。

想定される原因

計測オプションの「プログラムステップ情報を出力する」のチェックが外され、「組込みステップ情報を出力する」がチェックされている状態で、以下の場合に発生します。
1. COBOLのCOPY句ファイルやC/C++のincludeファイル等の組込みファイルがソース中で使用されていない。
2. 組込みファイルの存在するフォルダをサーチパスオプションで指定してない。

対処方法

組込みファイルを使用している場合は、サーチパスオプションで指定してください。

識別文字列は、最大件数件以上指定できません。

Java自動生成ソース及び、Visual Basic 6.0ソースの手書き部分・自動生成部分を判定する識別子は、100件以上指定できません。

想定される原因

Java自動識別オプションダイアログボックスで100件を超える識別子を指定しようとした場合に発生します。
VB6.0自動識別オプションダイアログボックスで100件を超える識別子を指定しようとした場合に発生します。

対処方法

100件以内で識別子を指定してください。

識別文字列の指定が重複しています。

Java自動生成ソース及び、Visual Basic 6.0ソースの手書き部分・自動生成部分を判定する識別子に既に同一の識別子が登録されています。

想定される原因

Java自動識別オプションダイアログボックスで、追加登録しようとした識別子は既に登録済みです。
VB6.0自動識別オプションダイアログボックスで、追加登録しようとした識別子は既に登録済みです。

対処方法

登録済み識別子の内容を確認してください。

識別子の組み合わせが不当です。

Java自動生成ソース及び、Visual Basic 6.0の計測時に、ソース内に識別子の不整合が存在します。

想定される原因

当メッセージは、Java自動識別オプションダイアログボックスまたは、VB6.0自動識別オプションダイアログボックスで［計測中に識別子不整合エラーを表示する］を選択した場合に表示されます。計測対象であるJava自動生成ソースまたは、Visual Basic 6.0ソース内に、以下の識別子不整合が存在する可能性があります。
1. 1つの開始識別子・終了識別子の範囲内に、入れ子状態の別の開始識別子・終了識別子が存在します。
2.開始識別子・終了識別子の組み合わせが不当です。
3.開始識別子のない終了識別子が存在します。
4.開始識別子と対になる終了識別子が存在しません。

対処方法

計測対象資産（Java自動生成ソース、Visual Basic 6.0）を確認し、必要であればソース内の識別子を修正します。

識別子ファイルの読み込みに失敗しました。

識別子ファイルの読み込みに失敗したため、識別子の取り込みができません。

想定される原因

Java自動識別オプションダイアログボックスまたは、VB6.0自動識別オプションダイアログボックスで識別子の取り込みをする際、識別子ファイルの読み込みで以下のエラーが発生した可能性があります。
1.識別子ファイルの内容エラー
2.識別子ファイルの排他エラー
3.AJDファイルを複数選択した場合のエラー

対処方法

読み込み対象の識別子ファイルを確認してください。

識別子ファイルの書き出しに失敗しました。

識別子ファイルの書き込みに失敗したため、識別子の書き出しができません。

想定される原因

Java自動識別オプションダイアログボックスまたは、VB6.0自動識別オプションダイアログボックスの書き出し処理で、識別子ファイルのアクセスで以下のエラーが発生した可能性があります。
1.識別子ファイルの排他エラー
2.識別子ファイル出力先フォルダの容量不足

対処方法

書き込み対象の識別子ファイルまたは、出力先フォルダを確認してください。

設定内容を消去し、外部ファイルの識別子を取り込みます。

識別子ファイルに格納される識別子を取り込むため、登録済みの識別子を消去します。

想定される原因

Java自動識別オプションダイアログボックスまたは、VB6.0自動識別オプションダイアログボックスで識別子取り込みの際に表示される確認メッセージです。

対処方法

登録済み識別子を消去し、識別子ファイルの識別子を取り込む場合は、処理を続行してください。

計測対象ファイル[ファイル名]のタグ表記([数値]行目開始)が正しくありません。

計測対象ソースファイル内のタグ表記に誤りがあります。

想定される原因

開始タグに対応する終了タグが発見されない状態で計測対象ソースファイルの終端に達した。

対処方法

計測対象ソースファイル内のタグ表記を確認してください。

このオプションを有効にすると、COBOLソースの計測がプログラムID単位となります。
また、計測結果のファイル名にプログラムID情報が付加されて表示されます。
副プログラムを計測したくない場合は、このオプションを無効にしてください。

計測オプションが変更されました。

想定される原因

計測オプション「プログラムID単位に計測する」を有効に変更した。

対処方法

プログラムID単位に計測されます。
メインプログラムのみを計測したい場合は、このオプションを無効にする必要があります。

このオプションを無効にすると、COBOLソースの計測はメインプログラムのみとなります。
また、計測結果のファイル名にプログラムID情報は付加されません。
副プログラムを含めて計測したい場合は、このオプションを有効にしてください。

計測オプションが変更されました。

想定される原因

計測オプション「プログラムID単位に計測する」を無効に変更した。

対処方法

メインプログラムのみ計測され、副プログラムは計測されません。
副プログラムも計測したい場合は、このオプションを有効にする必要があります。

このオプション設定時は、「フォルダ指定の場合、ヘッダーファイル(*.h)も計測する」機能は使用できません。

組込みメンバを計測する場合、「フォルダ指定の場合、ヘッダーファイル(*.h)も計測する」機能が使用できなくなります。

想定される原因

計測オプション「組込みメンバを計測する」を有効に変更した。

対処方法

組込みメンバを計測する場合、C/C++言語種別のフォルダ計測すると、組込みメンバステップ情報としてヘッダーファイル(*.h)が計測されます。
プログラムステップ情報としてヘッダーファイル(*.h)を計測したい場合は、このオプションを無効にする必要があります。

古いバージョンの計測結果ファイルが読み込まれました。プログラムID単位に計測する場合は、計測オプションの設定を見直してください。

計測結果ファイルのバージョンが古いため、計測オプション「プログラムID単位に計測する」を無効に変更します。

想定される原因

COBOL言語種別で計測した、古いバージョンの計測結果ファイルを読み込んだ。

対処方法

計測オプション「プログラムID単位に計測する」を無効に変更します。
プログラムID単位に計測したい場合は、オプションを有効に変更し、全計測する必要があります。

古いバージョンの計測結果ファイルが読み込まれました。
ヘッダーファイル(*.h)をプログラムステップ情報として計測する場合は、計測オプションの設定を見直してください。

計測結果ファイルのバージョンが古いため、計測オプション「フォルダ指定の場合、ヘッダーファイル(*.h)も計測する」を無効に変更します。

想定される原因

C/C++言語種別で計測した、古いバージョンの計測結果ファイルを読み込んだ。

対処方法

計測オプション「フォルダ指定の場合、ヘッダーファイル(*.h)も計測する」を無効に変更します。
ヘッダーファイル(*.h)も計測したい場合は、オプションを有効に変更し、全計測する必要があります。

サンプルの使い方

STEPCOUNTERのインストールフォルダには"sample"フォルダが作成され、各種言語のサンプルソースが格納されています。

ファイル名	文字コード	説明
sample.c	ShiftJIS	Cソース例
sample.cob	ShiftJIS	COBOLソース例
sample.java	ShiftJIS	JAVA(SQLJ)ソース例
生成sample.java	ShiftJIS	生成されたJAVA(SQLJ)ソース例
sample.idl	ShiftJIS	IDLソース例
sample.html	ShiftJIS	HTMLソース例
sample.vb, sample.cs, sample.asp, sample.aspx	ShiftJIS	VB.NET/C#/ASP/ASP.NETソース例
sample.frm	ShiftJIS	VB6.0ソース例

これらのソースは、ステップ数計測基準の説明例と同じものです。

ステップ数を計測するには

1. [ファイル]メニューの[新規計測]コマンドを選びます。
2. 新規計測 1/2 ダイアログボックスで[言語種別]、[対象資産の指定方法]及び、[文字コードの指定]にShiftJISを選択し、[次へ]ボタンを押下します。
3. 新規計測 2/2 ダイアログボックスで計測対象となるソースファイルまたは、ソースファイルが存在するフォルダを指定します。
4. 全ての計測対象の指定が済んだら、[完了]ボタンを押下すると計測処理が開始され、計測結果を画面に表示します。

表示形式を変更するには

1. [オプション]メニューの[表示オプション]コマンドを選びます。
2. 表示オプション ダイアログボックスの［初期画面表示］で[帳票形式]か[リストビュー形式]を選択します。
3. リストビュー形式を選択した場合は、同画面で初期表示する項目を選択することができます。
4. 帳票形式を選択した場合は、「帳票」タブを選択して帳票オプション ダイアログボックスを表示することにより、表示する情報を選択することができます。
5. [計測]メニューの[変更部分のみ計測]コマンドを選ぶことにより、再度計測することなく表示形式を変更することができます。
6. [表示]メニューから各種表示項目の表示／非表示などが選択できます。詳細については[表示]メニューのコマンドを参照してください。

計測結果を印刷するには

1. 計測結果を印刷する場合は、帳票形式で表示してください。リストビュー形式では印刷はできません（後で説明するCSV形式でファイル保存して、表計算アプリケーションなどで印刷してください）。
2. [ファイル]メニューの[印刷]コマンドを選びます。
3. 印刷 ダイアログボックスにて必要項目の設定をし、[OK]ボタンを押下します。
4. 計測結果がプリンタに印刷されます。
5. 詳細については計測結果の印刷方法を参照してください。

計測結果をCSVファイルに保存するには

1. [ファイル]メニューの[CSV形式ファイルの作成]コマンドを選びます。
2. [ファイルの保存]ダイアログボックスより、CSV形式ファイル名を指定し[OK]ボタンを押下します。
3. 指定したファイル名でCSV形式ファイルが作成されます。

リストビュー形式で表示している場合は、任意の行を選択してその行の情報だけを出力できます。

1. CSV形式ファイルへ出力する計測結果をリストビューより選択します（複数選択可能）。
2. [ファイル]メニューの[選択範囲のみCSV形式ファイルの作成]コマンドを選びます。
3. [ファイルの保存]ダイアログボックスより、CSV形式ファイル名を指定し[OK]ボタンを押下します。

CSVファイルに保存する場合、表示形式によって保存される書式が異なります。帳票形式の場合またはリストビュー形式の場合を参照してください。

計測結果を保存するには

1. [ファイル]メニューの[名前を付けて保存]コマンドを選びます。
2. [ファイルの保存]ダイアログボックスより、計測結果ファイル名を指定し[OK]ボタンを押下します。
3. 指定したファイル名で計測結果ファイルが作成されます。

対象資産を変更した場合や計測結果が更新された場合は、[ファイル]メニューの[上書き保存]コマンドを選びます。

計測結果ファイルを読み込むには

1. [ファイル]メニューの[読み込み]コマンドを選びます。
2. [ファイルを開く]ダイアログボックスより、計測結果ファイルを選択し[OK]ボタンを押下します。
3. 計測結果を画面に表示します。

注意　計測対象ソースでコピー句ファイルやインクルードファイルを使用している場合は、計測の前にサーチパスオプションの設定が必要です。

組込みステップ数

組込みステップ数は、インクルードファイル/登録集/自動生成部のステップ数です。組込みステップ数としてカウントする情報は言語種別により異なりますので、下記の表を参照して確認してください。

言語種別	説明
COBOL	登録集のステップ数　（注１）
C/C++	インクルードファイルのステップ数　（注１）
Java	出力されません
IDL	出力されません
Java自動生成ソース	自動生成部のステップ数　（注２）
HTML/JSP/JS/CSS	出力されません
.NET（ASP/C#/VB）/HTML/JS/CSS	出力されません
Visual Basic 6.0	自動生成部のステップ数　（注２）

※形式は、総ステップのトピックを参照してください。

注１）組込みメンバステップ情報（帳票形式、リストビュー形式）の出力が可能です。
注２）組込みメンバステップ情報（帳票形式、リストビュー形式）の出力はできません。

注記項目

出力項目	説明
注記項目	帳票オプションにて設定した文字列

平均

出力項目	説明
平均	全計測対象資産の合計を計測本数で平均した値

出力項目	説明
合計	全計測対象資産の合計

小計

出力項目	説明
小計	同一フォルダ単位での合計（帳票オプションの[ファイルパス名を出力する]指定時）

組込み形態別

組込み形態別は、登録集の呼出し形態毎、またはインクルードファイルの呼出し形態毎の組込みメンバのステップ数の合計です。出力する情報は言語種別により異なりますので、下記の表を参照して確認してください。

【COBOLの場合】

出力項目	説明
C (COPY)	COPY句の呼出しで組み込んだメンバ (登録集) のステップ数の合計
C率	組込みステップ数の中の C (COPY) の割合　(注 1)
I (INC)	INCLUDE句の呼出しで組み込んだメンバ (登録集) のステップ数の合計
I率	組込みステップ数の中の I (INC) の割合　(注 1)

【C/C++の場合】

出力項目	説明
I (inc)	include呼出しで組み込んだメンバ (インクルードファイル) のステップ数の合計
I率	組込みステップ数の中の I (inc) の割合　(注 1)
R (rc)	rcinclude呼出しで組み込んだメンバ (インクルードファイル) のステップ数の合計
R率	組込みステップ数の中の R (rc) の割合　(注 1)

※「Java」、「IDL」、「Java自動生成ソース」、「HTML/JSP/JS/CSS」及び、「.NET（ASP/C#/VB）/HTML/JS/CSS」の時は、出力されません。

注 1 ）リストビュー形式の場合は出力されません。

手書きステップ数

手書きステップ数は、インクルードファイル/登録集を除くステップ数、または手書き部のステップ数です。手書きステップ数としてカウントする情報は言語種別により異なりますので、下記の表を参照して確認してください。

言語種別	説明
COBOL	登録集を除くステップ数
C/C++	インクルードファイルを除くステップ数
Java	出力されません
IDL	出力されません
Java自動生成ソース	手書き部のステップ数
HTML/JSP/JS/CSS	出力されません
.NET（ASP/C#/VB）/HTML/JS/CSS	出力されません
Visual Basic 6.0	手書き部のステップ数

※形式は、総ステップのトピックを参照してください。

総ステップ数

総ステップ数は、１プログラムを構成するステップ数です。総ステップ数として出力する情報は言語種別により異なりますので、下記の表を参照して確認してください。

【COBOLの場合】

出力項目	説明
宣言	宣言部の有効ステップ数　（プログラムの先頭からPROCEDURE DIVISION直前までのステップ数）
宣言率	合計ステップ数の中の宣言ステップ数の割合　(注１)
SQL　(宣言部)	宣言部の埋め込みSQLのステップ数
SQL率　(宣言部)	宣言ステップ数の中のSQLの割合
ETC　(宣言部)	SQLを除いた宣言ステップ数
ETC率　(宣言部)	宣言ステップ数の中のETCの割合　(注１)
実行	実行部の有効ステップ数　（PROCEDURE DIVISION以降のステップ数）
実行率	合計ステップ数の中の実行ステップ数の割合　(注１)
SQL　(実行部)	実行部の埋め込みSQLのステップ数
SQL率　(実行部)	実行ステップ数の中のSQLの割合
ETC　(実行部)	SQLを除いた実行ステップ数
ETC率　(実行部)	実行ステップ数の中のETCの割合　(注１)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
計	宣言部のステップ数と実行部のステップ数と注釈ステップ数の合計

【C/C++の場合】

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注１)
SQL	埋め込みSQLのステップ数
SQL率	有効ステップ数の中のSQLの割合
ETC	SQLを除いた有効ステップ数
ETC率	有効ステップ数の中のETCの割合　(注１)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
計	有効ステップ数と注釈ステップ数の合計

【Java, Java自動生成ソースの場合】

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注１)

SQL	埋め込みSQLのステップ数
SQL率	有効ステップ数の中のSQLの割合
ETC　(有効行)	SQLを除いた有効ステップ数
ETC率　(有効行)	有効ステップ数の中のETCの割合　(注 1)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
Javadoc	Javadocのステップ数
Javadoc率	注釈ステップ数の中のJavadocの割合
ETC　(注釈行)	Javadocを除いた注釈のステップ数
ETC率　(注釈行)	注釈ステップ数の中のETCの割合　(注 1)
計	有効ステップ数と注釈ステップ数の合計

【IDLの場合】

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注 1)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
計	有効ステップ数と注釈ステップ数の合計

【HTML/JSP/JS/CSSの場合】

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注 1)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
計	有効ステップ数と注釈ステップ数の合計

【.NET（ASP/C#/VB）/HTML/JS/CSSの場合】

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注 1)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
計	有効ステップ数と注釈ステップ数の合計

【VB 6.0 の場合】

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注 1)
注釈	注釈のステップ数

注釈率	合計ステップ数の中の注釈の割合
計	有効ステップ数と注釈ステップ数の合計

注１） リストビュー形式の場合は出力されません。

更新日付

出力項目	説明
更新日付	計測対象ファイルの更新日付。（帳票オプションの[ファイルの更新日付を出力する]指定時）

プログラム名

出力項目	説明
プログラム名	計測対象ファイル

注意　ファイル名の出力可能な長さは言語種別や帳票オプション（SQLステップ数の出力、ファイル日付の出力）により変化します。各状況による最大長以後のファイル名は出力されません。

シーケンス番号

出力項目	説明
シーケンス番号	ソース／インクルードファイル（登録集ファイル）に対する連番

フォルダ名

出力項目	説明
フォルダ名	計測対象ファイルのフォルダ名

ファイル情報

出力項目	説明
ファイル名	計測対象ファイル名
種類	計測対象ファイルの言語名 詳細は「新規計測の方法」の注意欄を参照
サイズ	計測対象ファイルの論理サイズ
更新日付	計測対象ファイルの更新日付

使用プログラム及び組込みメンバ名

使用プログラム及び組込みメンバ名は、登録集を展開しているプログラム名/登録集名、または include展開しているプログラム名/インクルード名です。出力する情報は言語種別により異なりますので、下記の表を参照して確認してください。

【COBOLの場合】

出力項目	説明
使用プログラム及び組込みメンバ名	(P) 展開しているプログラム名 (C) 展開している登録集名

【C/C++の場合】

出力項目	説明
使用プログラム及び組込みメンバ名	(P) 展開しているプログラム名 (I) 展開しているインクルード名

注意　表示されるファイル名は１２文字までです。１３文字以上のファイル名は１２文字目までしか表示されません。

使用個所

出力項目	説明
私用回数	計測対象内での使用回数

のべ総ステップ数

出力項目	説明
のべ総ステップ	計測対象内でののべステップ数（インクルード（登録集）ステップと使用回数との積）

使用個所

出力項目	説明
使用箇所	登録集の展開箇所（Ｃでは未計測）

ステップ数内訳

ステップ数内訳は、組込みメンバの計測ステップ数です。

出力項目	説明
有効	注釈を除いたステップ数
有効率	合計ステップ数の中の有効ステップ数の割合　(注１)
SQL	埋め込みSQLのステップ数
SQL率	有効ステップ数の中のSQLの割合
ETC	SQLを除いた有効ステップ数
ETC率	有効ステップ数の中のETCの割合　(注１)
注釈	注釈のステップ数
注釈率	合計ステップ数の中の注釈の割合
計	有効ステップ数と注釈ステップ数の合計

注１）リストビュー形式の場合は出力されません。

組込みメンバ名

出力項目	説明
組込みメンバ名	インクルード（登録集）ファイル名

注意　ファイル名の出力可能な長さは言語種別や帳票オプション（SQLステップ数の出力、ファイル日付の出力）により変化します。各状況による最大長以後のファイル名は出力されません。

SIMPLIA/MF-STEPCOUNTER
Javaソフトウェアメトリクス計測機能
V50L42

オンラインマニュアル

第 1.0 版

はじめに

SIMPLIA/MF-STEPCOUNTER　Javaソフトウェアメトリクス計測機能は、Java開発資産の計測をおこなうアプリケーションです。当機能を使用することにより、クラスやメソッドの開発進捗状況や設計品質を把握することができます。

新機能

・JDK(R) 5.0に対応しました。
　JDK1.3/1.4版とJDK5.0版では起動するメニュー（batファイルまたはjarファイル）が異なりますのでご注意ください。

ヘルプを読むために

HTML3.2をサポートするWWWブラウザ（インターネットエクスプローラ V3.02以降、Netscape NavigatorV4.03以降）をお使いください。

登録商標について

略記について

本オンラインマニュアルでは、各製品を次のように略記しています。

「Microsoft(R) Windows(R) 2000 Professional operating system」	→	「Windows 2000」または、「Windows 2000 Professional」
「Microsoft(R) Windows(R) 2000 Server operating system」	→	「Windows 2000」または、「Windows 2000 Server」
「Microsoft(R) Windows(R) 2000 Advanced Server operating system」	→	「Windows 2000」または、「Windows 2000 Advanced Server」
「Microsoft(R) Windows(R) XP Professional 」	→	「Windows(R) XP」または、「Windows (R) XP Professional」
「Microsoft(R) Windows Vista(R) Enterprise Edition 」	→	「Windows Vista」
「Microsoft(R) Windows Server(R) 2003, Standard Edition」 「Microsoft(R) Windows Server(R) 2003 R2, Standard Edition」	→	「Windows Server 2003」
「Microsoft(R) Windows Server(R) 2003, Enterprise Edition」 「Microsoft(R) Windows Server(R) 2003 R2, Enterprise Edition」	→	「Windows Server 2003」
「Microsoft(R) Windows Server(R) 2008 Standard」 「Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM)」 「Microsoft(R) Windows Server(R) 2008 Enterprise」 「Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM)」	→	「Windows Server 2008」
「Java(TM)」	→	「Java」
「Java(TM) 2 SDK, Standard Edition Version 1.3」	→	「JDK」 または、 「JDK1.3」
「Java(TM) 2 SDK, Standard Edition Version 1.4」	→	「JDK」 または、 「JDK1.4」
「Java(TM) 2 Platform Standard Edition Development Kit 5.0」	→	「JDK」 または、 「JDK5.0」

1.はじめに
　1.2.動作環境

ハードウェア条件

操作マシン
Windows 2000,Windows XP,Windows Vista, Windows Server 2003またはWindows Server 2008が動作するマシン

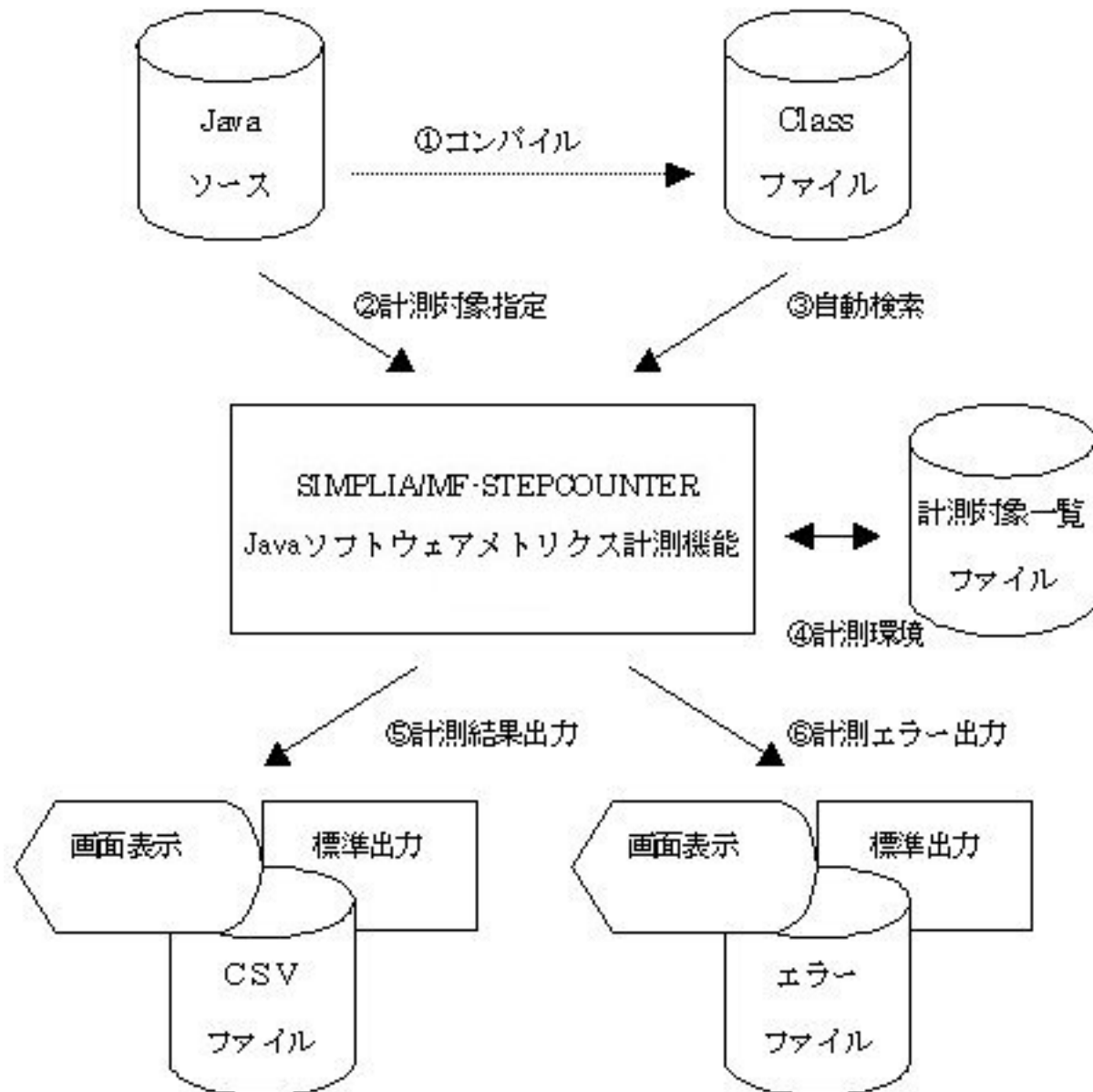
ソフトウェア条件

必須ソフトウェア		
ソフトウェア名称	適用要件（バージョン／レベル）	備考
OS	Windows 2000 Windows XP Windows Vista Windows Server 2003 Windows Server 2008	左記のいずれかのOSが必要です。
JDK	JDK1.3/1.4/5.0	JDKのbinフォルダが環境変数pathに追加されている必要があります。

SIMPLIA/MF-STEPCOUNTER　Javaソフトウェアメトリクス計測機能

1.はじめに
　1.3.動作概要

動作概要

当ツールは、Java開発資産であるJavaソースと、Javaソースのコンパイル結果であるClassファイルを入力とし、計測をおこなうツールです。計測は、クラス単位の計測とメソッド単位の計測が可能で、計測結果は指定の計測単位ごとに出力されます。計測結果は、画面上への表示の他、CSVファイルにも出力されます。一度計測した資産は、計測対象一覧ファイルとして保存することで、何度でも同じ環境で計測することができます。ただし、SQLJを使用したJavaソースは計測できません。

①	コンパイル	当ツール実行前に、事前にJavaソースをコンパイルする必要があります。
②	計測対象指定	計測対象となるJavaソースをファイルまたはフォルダ単位に指定します。指定資産のうち、拡張子が.Javaのファイルのみを計測対象とし、計測をおこないます。

③	自動検索	Classファイルの計測は、Javaソースの解析によりファイルを特定し、環境変数CLASSPATHに指定されるパス名とツール内に指定されるクラスパス名から自動的に検索し、計測をおこないます。
④	計測環境	指定の環境（出力先ＣＳＶファイル名／計測対象資産名／計測単位／クラスパス名／アラーム指定形式）を保存できます。保存した計測対象一覧ファイルを使用することで同一環境で何度でも計測することが可能です。
⑤	計測結果出力	指定の計測単位ごとに計測結果を出力します。
⑥	計測エラー出力	計測エラー発生時は、エラー内容を出力します。

計測結果として出力される項目は以下のとおりです。
(1)クラス単位の計測を指定した場合

No.	計測項目	情報取得元	内容
1	クラス名	Javaソース	計測単位となるクラス名を出力します。
2	文数	Javaソース	クラス内の実行文の数を計測します。
3	継承数	Classファイル	クラスの階層数を計測します。
4	フィールド変数	Classファイル	クラス内のフィールド変数の数を計測します。
5	インスタンス変数	Classファイル	クラス内のインスタンス変数の数を計測します。
6	パブリック変数	Classファイル	クラス内のパブリック変数の数を計測します。
7	メソッド数	Classファイル	クラス内に定義されるメソッドの数を計測します。
8	インスタンスメソッド数	Classファイル	クラス内に定義されるインスタンスメソッド数を計測します。
9	パブリックメソッド数	Classファイル	クラス内に定義されるパブリックメソッド数を計測します。
10	定義ファイル名	Javaソース	「1.クラス名」が定義されるJavaソースのファイル名を出力します。Javaソース内に複数のクラスが定義される場合、Javaソース上の先頭クラスにのみ出力され、その他のクラスには出力されません

(2)メソッド単位の計測を指定した場合

No.	計測項目	情報取得元	内容
1	メソッド名	Javaソース	計測単位となるクラス名を出力します。
2	文数	Javaソース	メソッド内の実行文の数を計測します。
3	分岐文数	Javaソース	メソッド内に定義される条件分岐文（ｉｆ文/switch内のcase文）の数を計測します。
4	ループ文数	Javaソース	メソッド内に定義されるループ文（for文/while文/do-while文）の数を計測します。
5	メソッドＣＡＬＬ数	Javaソース	他メソッドの実行数を計測します。
6	定義ファイル名	Javaソース	「1.メソッド名」が定義されるJavaソースのファイル名を出力します。Javaソース内に複数のメソッドが定義される場合、Javaソース上の先頭メソッドにのみ出力され、その他のメソッドには出力されません。

2.起動方法

2.1.コマンドライン引数

コマンドライン引数には、以下のパラメタが存在します。各パラメタの指定順は任意とします。
ー 起動形態パラメタ
ー 対象資産パラメタ
ー クラスパスパラメタ
ー オプションパラメタ
ー ＣＳＶパラメタ
ー ＥＲＲパラメタ
パラメタ間には、排他関係にあるパラメタも存在するため、指定には注意が必要です。
各パラメタの指定方法に記述される記号は、以下の内容を指します。

記号	説明
[]	省略可能です。
{}	複数のパラメタから１つ選択し、指定します。
下線	デフォルト値です。省略時は、デフォルト値が指定されたものとします。

2.起動方法
2.1.コマンドライン引数
2.1.1.－起動形態パラメタ

指定方法

```
              -b
[ {              } ]
              -f
```

説明

当ツールの起動方法を指定します。起動形態パラメタ省略時は、-fが指定されたものとみなします。

指定内容	説明
-b	バッチ処理を起動します。当パラメタ指定時は、対象資産パラメタの指定が必要です。
-f	画面処理を起動します。当パラメタ指定時は、対象資産／クラスパス／オプション／ＣＳＶパラメタは、指定できません。

2.起動方法
2.1.コマンドライン引数
2.1.2.対象資産パラメタ

指定方法

［｛ -l 計測対象一覧ファイル名
-d フォルダ名
ファイル名 ｝］

説明

計測対象となる対象資産を指定します。

指定内容	説明
-l（エル） 計測対象一覧ファイル名	計測対象一覧ファイルに格納される内容を計測対象とします。計測対象一覧ファイルは、対象資産／クラス／ＣＳＶ出力先／計測オプションを格納したファイルです。当パラメタが指定される場合、クラス／オプション／ＣＳＶパラメタは指定できません。
-d フォルダ名	指定フォルダ名配下に存在する全ファイルを計測対象とします。(サブフォルダを含めて計測する場合はオプションパラメタに-sを指定します。)
ファイル名	計測対象となるJavaファイル名を指定します。当パラメタには、ワイルドカード指定が可能です。ワイルドカード指定時は、該当するJavaファイルの計測をおこないます。

2.起動方法
2.1.コマンドライン引数
2.1.3.クラスパスパラメタ

指定方法

```
                    フォルダ名                          フォルダ名
[   -c   {                         }   [  ;  {                         }  ]   …  ]
                    Jarファイル名                       Jarファイル名
```

説明

当ツールでは、クラス単位の計測時に対象資産として指定されるJavaファイルからClass名を特定し、Classファイルを検出し計測します。Classファイル検出時の検索パス（フォルダ名またはJarファイル名）を指定します。

指定内容	説明
フォルダ名	Classファイル検索対象となるフォルダ名を指定します。
Jarファイル名	Classファイルが格納されるJarファイル名を指定します。

複数の検索パスを指定する場合、パス名間をセミコロン（；）で区切り、継続して指定します。環境変数CLASSPATHが存在する場合、環境変数CLASSPATHへの指定パスも検索対象となるため、環境変数CLASSPATHの内容は、当パラメタに指定する必要はありません。また、環境変数CLASSPATHに必要な検索パスがすべて設定されている場合や、メソッド単位の計測をおこなう場合は、当パラメタの指定は必要ありません。
Classファイルは、環境変数CLASSPATHへの指定順、当パラメタでの指定順で検索されます。

2.起動方法

2.1.コマンドライン引数

2.1.4.オプションパラメタ

指定方法

[-m] [-s]

説明

計測オプションを指定します。

指定内容	説明
-m	メソッド単位の計測を指定します。当パラメタ省略時は、クラス単位の計測を実行します。当パラメタが指定される場合、クラスパスパラメタの指定を無効とします。
-s	当パラメタは、対象資産パラメタにフォルダ名が指定される場合に有効となります。当パラメタが指定される場合、対象資産（フォルダ）のサブフォルダに含まれるJavaファイルも計測対象とします。対象資産パラメタにファイル名が指定される場合、当パラメタの指定を無効とします。

2.起動方法
　2.1.コマンドライン引数
　　2.1.5.ＣＳＶパラメタ

指定方法

［　　-csv　　ＣＳＶファイル名　　］

説明

計測結果の出力先を指定します。

指定内容	説明
ＣＳＶファイル名	計測結果の出力先となるＣＳＶファイル名を指定します。当パラメタ省略時は、計測結果を標準出力に出力します。

2.起動方法
　2.1.コマンドライン引数
　　2.1.6.ＥＲＲパラメタ

指定方法

[　　　　　　-efile　　　　　　]

説明

計測エラーの出力先を指定します。

指定内容	説明
-efile	当パラメタ指定時は、計測中に発生するエラーをエラーファイルに出力します。エラーファイルは、当ツールのインストールフォルダ配下のＴｅｍｐフォルダに「mfstpjm.err」のファイル名で出力されます。当パラメタを省略した場合、計測中に発生したエラーは、標準出力に出力されます。また、当パラメタを指定した場合でも、エラーファイルへの出力異常を検出した場合は、計測中に発生したエラーを、標準出力に出力します。

2.起動方法

2.2.パラメタの排他関係

各パラメタの排他関係を以下に示します。

パラメタ		ＥＲＲ	ＣＳＶ	オプション		クラスパス	対象資産			起動形態	
		-efile	-csv～	-s	-m	-c～	ファイル名	-d～	-l～	-f	-b
起動形態	-b	○	○	○	○	○	●（どれか１つ）			×	×
	-f	○	×	×	×	×	×	×	×	×	×
対象資産	-l～	○	×	×	×	×	×	×	×	－	－
	-d～	○	○	○	○	○	×	×	－	－	－
	ファイル名	○	○	△	○	○	×	－	－	－	－
クラスパス	-c～	○	○	○	○	×	－	－	－	－	－
オプション	-m	○	○	○	×	△	－	－	－	－	－
	-s	○	○	×	－	－	－	－	－	－	－
ＣＳＶ	-csv～	○	×	－	－	－	－	－	－	－	－
ＥＲＲ	-efile	×	－	－	－	－	－	－	－	－	－

●：指定必須、○：指定可、△：指定無効、×：指定不可

2.起動方法
　2.3.ツールの起動

当ツールの起動は、以下の手順で実施してください。

(1) 当ツールのインストールフォルダに格納されているmfstpjm.jar(JDK1.3/1.4版)またはmfstpjm5.jar(JDK5.0版)ファイルを環境変数CLASSPATHに追加します。

追加例(JDK1.3/1.4版)：
set CLASSPATH=［インストールフォルダ］¥mfstpjm.jar;%CLASSPATH%

追加例(JDK5.0版)：
set CLASSPATH=［インストールフォルダ］¥mfstpjm5.jar;%CLASSPATH%

(2) 当ツールのインストールフォルダをカレントフォルダとし、java.exeにより当ツールの起動クラスを実行します。当ツールの起動クラス名は、com.fujitsu.simplia.StepCounter.metricsです。起動に必要なパラメタがある場合は、起動時に合わせて指定してください。パラメタの内容は、「2.1.コマンドライン引数」を参照してください。

起動例：
cd [インストールフォルダ]
java com.fujitsu.simplia.StepCounter.metrics [パラメタ]

注）当ツールの実行前に事前にJDKのbinフォルダを環境変数pathに追加する必要があります。

3.画面説明

3.1.メイン画面

レイアウト

説明

計測処理に必要な以下の項目の指定をおこないます。

必須指定内容	説明
ＣＳＶファイル名	計測結果の出力先ファイル名を指定します。
対象資産一覧	計測対象となる資産（フォルダ／Javaファイル）名を指定します。

メニューコマンド説明

（1）ファイルメニュー

メニューアイテム	説明
新規作成(N)	新しく計測をおこない、新規の計測対象一覧ファイルを作成します。
計測対象一覧ファイルを開く...(O)	既存の計測対象一覧ファイルを読み込みます。
上書き保存(S)	計測対象一覧ファイルの内容を更新して保存します。
名前を付けて保存...(A)	計測対象一覧ファイルの名前を指定して保存します。
SIMPLIA/MF-STEPCOUNTERの終了(X)	当ツールを終了します。

（2）計測メニュー

メニューアイテム	説明
計測(K)	計測処理を実行し、計測結果を表示します。

（３）表示メニュー

メニューアイテム	説明
計測結果表示(D)	計測結果を画面表示します。

（４）オプションメニュー

メニューアイテム	説明
クラスパス指定...(P)	Classファイルの検索パスを指定します。
計測オプション指定...(I)	計測時のオプションを指定します。
アラーム指定...(L)	計測結果のアラーム表示内容を指定します。

（５）ヘルプメニュー

メニューアイテム	説明
MF-STEPCOUNTER ヘルプ(T)	ヘルプ画面を表示します。
バージョン情報...(A)	当ツールのバージョンレベルを表示します。

画面コントロール説明

画面コントロール	説明
「ＣＳＶファイル名」テキストフィールド	計測結果の出力先ＣＳＶファイル名を入力します。「ＣＳＶファイル参照」ボタンにより、ファイルの選択が可能です。
「ＣＳＶファイル参照」ボタン	計測結果の出力先ＣＳＶファイル名を選択します。
「対象資産形式」コンボボックス	計測対象資産の資産形式を選択します。
「対象資産名」テキストフィールド	計測対象となる資産名を入力します。対象資産形式によっては、フォルダの選択も可能です。
「対象資産参照」ボタン	計測対象となる対象資産名を選択します。選択内容は、「対象資産形式」コンボボックスでの資産形式の選択内容により異なります。詳細は、「4.2.2.対象資産の指定方法」を参照してください。
「対象資産一覧」リスト	選択済の対象資産名を表示します。
「追加」ボタン	「対象資産名」テキストフィールドに指定された内容を、「対象資産一覧」リストの末尾に追加します。
「削除」ボタン	「対象資産一覧」リストの選択資産を、「対象資産一覧」リストから削除します。
「全て削除」ボタン	「対象資産一覧」リストに表示される全資産名を、「対象資産一覧」リストから削除します。
「計測」ボタン	計測処理を実行し、計測結果を表示します。

3.画面説明
　3.2.計測中断ダイアログボックス

レイアウト

説明

計測状況を表示します。

画面コントロール説明

画面コントロール	説明
「計測率」ラベル	メイン画面で指定する対象資産の計測率を表示します。
「ファイル名」ラベル	計測中のJavaファイル名を表示します。
「計測状況」キャンパス	計測状況が描画されます。
「中断」ボタン	計測中のJavaファイルの計測終了後に処理を中断します。

レイアウト

説明

計測結果を表形式で表示します。表示データ内容は、計測単位の指定により異なります。

（１）クラス単位の計測時

表示データ項目	説明
クラス名	計測単位となるクラス名を表示します。
文数	クラス内の文数を表示します。
継承数	クラスの継承数を表示します。
フィールド変数の数	クラス内に定義されるフィールド変数の数を表示します。
インスタンス変数の数	クラス内に定義されるインスタンス変数の数を表示します。
パブリック変数の数	クラス内に定義されるパブリック変数の数を表示します。
メソッド数	クラス内に定義されるメソッド数を表示します。
インスタンスメソッド数	クラス内に定義されるインスタンスメソッド数を表示します。
パブリックメソッド数	クラス内に定義されるパブリックメソッド数を表示します。
定義ファイル名	クラスが定義されるJavaファイル名を表示します。Javaファイル内の先頭に定義されるクラス名にのみ出力されます。

（２）メソッド単位の計測時

表示データ項目	説明
メソッド名	計測単位となるメソッド名表示します。
文数	メソッド内の文数を表示します。
条件分岐文数	メソッド内に定義される条件分岐文数を表示します。
ループ数	メソッド内に定義されるループ文数を表示します。
メソッドＣＡＬＬ数	メソッド内に定義されるメソッドＣＡＬＬ文数を表示します。
定義ファイル名	メソッドが定義されるJavaファイル名を表示します。Javaファイル内の先頭に定義されるメソッド名にのみ出力されます。

メニューコマンド説明

（１）ファイルメニュー

メニューアイテム	説明
終了(X)	計測結果表示画面を終了します。

（２）オプションメニュー

メニューアイテム	説明
アラーム指定...(L)	計測結果のアラーム表示内容を指定します。

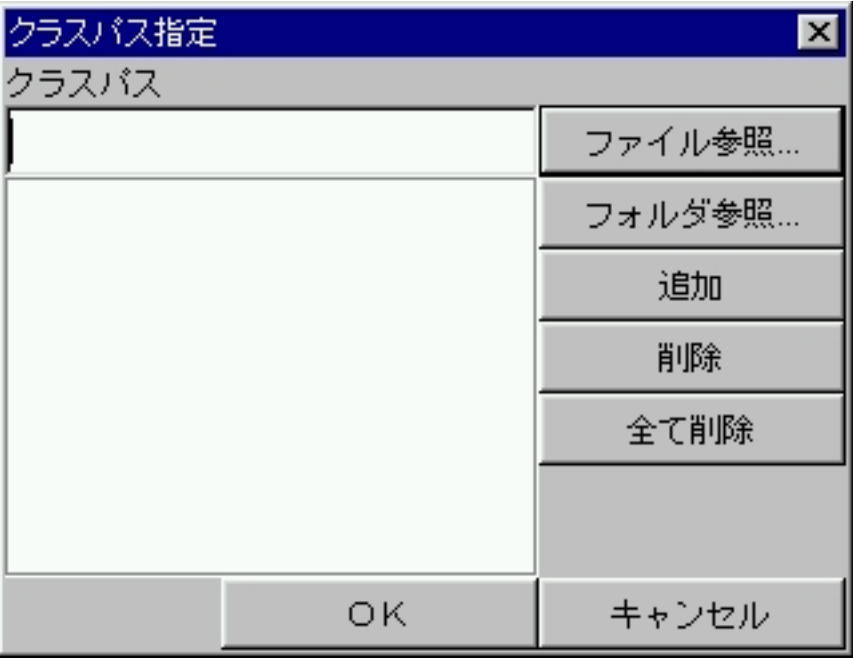
（３）ヘルプメニュー

メニューアイテム	説明
MF-STEPCOUNTER ヘルプ(T)	ヘルプ画面を表示します。
バージョン情報...(A)	当ツールのバージョンレベルを表示します。

画面コントロール説明

画面コントロール	説明
「計測結果表」テーブル	計測結果を表示します。アラーム指定ダイアログボックスでのアラーム表示指定により、以下のフォントカラーで表示されます。 －上限値以上の計測値：赤 －下限値以下の計測値：青 －上記以外：黒

3.画面説明
　3.4.クラスパス指定ダイアログボックス

レイアウト

説明

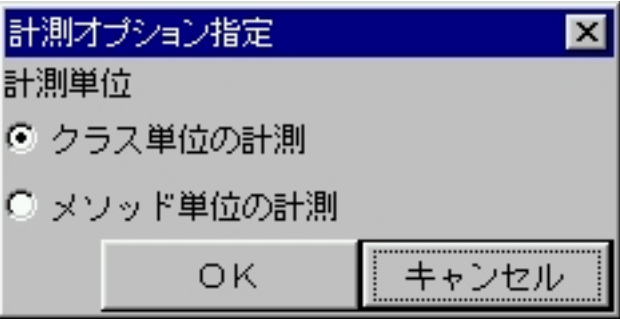
Classファイル計測時の検索パスとなるクラスパス名（フォルダ名／Ｊａｒファイル名）を指定します。

画面コントロール説明

画面コントロール	説明
「クラスパス名」テキストフィールド	クラスパスとなるフォルダ名・Jarファイル名の入力をおこないます。「ファイル参照」ボタン・「フォルダ参照」ボタンにより、フォルダ・Jarファイルの選択が可能です。
「ファイル参照」ボタン	クラスパスとなるJarファイル名を選択します。
「フォルダ参照」ボタン	クラスパスとなるフォルダ名を選択します。
「クラスパス一覧」リスト	選択済のクラスパス名を表示します。
「追加」ボタン	「クラスパス名」テキストフィールドに指定された内容を、「クラスパス一覧」リストの末尾に追加します。
「削除」ボタン	「クラスパス一覧」リストの選択資産を、「クラスパス一覧」リストから削除します。
「全て削除」ボタン	「クラスパス一覧」リストに表示される全パス名を、「クラスパス一覧」リストから削除します。
「ＯＫ」ボタン	指定した内容を保存し、当ダイアログボックスを終了します。
「キャンセル」ボタン	指定した内容を無効とし、当ダイアログボックスを終了します。

3.画面説明

3.5.オプション指定ダイアログボックス

レイアウト

説明

計測単位（クラス単位の計測／メソッド単位の計測）を指定します。

画面コントロール説明

画面コントロール	説明
「クラス単位の計測」ラジオボタン	対象資産となるJavaファイルに定義されるクラス単位に計測処理を実行します。
「メソッド単位の計測」ラジオボタン	対象資産となるJavaファイルに定義されるメソッド単位に計測処理を実行します。
「ＯＫ」ボタン	指定した内容を保存し、当ダイアログボックスを終了します。
「キャンセル」ボタン	指定した内容を無効とし、当ダイアログボックスを終了します。

3.画面説明
　3.6.アラーム指定ダイアログボックス

レイアウト

説明

計測結果表示画面で表示するアラーム範囲の指定をおこないます。当ダイアログボックスで指定される上限値以上の計測値／下限値以下の計測値をアラーム表示の対象とします。

画面コントロール説明

画面コントロール	説明
「パーセンタイル」ラジオボタン	パーセンタイル形式でアラームを表示します。 詳細は「7.3.パーセンタイル形式のアラーム表示」を参照してください。
「閾値」テキストフィールド	アラーム表示対象となる下限の閾値設定を行います。下限の閾値は（０％～５０％）の範囲で指定します。当フィールドが空白の場合、および、０％が指定された場合、アラーム表示は行いません。当指定は、「パーセンタイル」ラジオボタンが選択された場合に有効となります。
「上限値／下限値」ラジオボタン	上限値／下限値形式でアラームを表示します。
「項目」コンボボックス	アラーム指定の対象となる計測項目を選択します。当指定は、「上限値／下限値」ラジオボタンが選択された場合に有効となります。
「上限値」テキストフィールド	アラーム表示対象となる上限値の設定をおこないます。下限値より大きい内容でなければなりません。当フィールドが空白の場合、上限値以上の計測値に対するアラーム表示はおこないません。当指定は、「上限値／下限値」ラジオボタンが選択された場合に有効となります。

「下限値」テキストフィールド	アラーム表示対象となる下限値の設定をおこないます。上限値より小さい内容でなければなりません。当フィールドが空白の場合、下限値以下の計測値に対するアラーム表示はおこないません。当指定は、「上限値／下限値」ラジオボタンが選択された場合に有効となります。
「デフォルト値」ボタン	パーセンタイル形式が選択され、閾値として25%が設定されます。
「ＯＫ」ボタン	指定した内容を保存し、当ダイアログボックスを終了します。
「キャンセル」ボタン	指定した内容を無効とし、当ダイアログボックスを終了します。

3.画面説明
　3.7.バージョン情報ダイアログボックス

レイアウト

説明

当ツールのツール名／機能名／バージョンレベル／著作権を表示します。

画面コントロール説明

画面コントロール	説明
「ＯＫ」ボタン	当ダイアログボックスを終了します。

3.画面説明
　3.8.ヘルプ表示画面

レイアウト

説明

当ヘルプを表示します。

4.操作手順
4.1.バッチ処理
4.1.1.ファイル指定による計測方法

概要

コマンドライン引数の指定で、指定のJavaファイル／Classファイルをバッチ処理で計測をおこないます。

操作手順

1. コマンドライン引数に以下の指定をおこない、当ツールを起動します。

   -b [Javaファイル名]

   実行例：
   c:¥Java¥test.javaを計測対象のJavaファイルとした場合
   java com.fujitsu.simplia.StepCounter.metrics -b c:¥Java¥test.java

4.操作手順
4.1.バッチ処理
4.1.2.フォルダ指定による計測方法

概要

コマンドライン引数の指定で、指定フォルダに含まれるJavaファイルと、Javaファイルに定義されるClassファイルを、バッチ処理で計測をおこないます。

操作手順

1. コマンドライン引数に以下の指定をおこない、当ツールを起動します。

   -b -d [フォルダ名]

   実行例：
   c:¥Javaを計測対象のフォルダとした場合
   java com.fujitsu.simplia.StepCounter.metrics -b -d c:¥Java

※引数内に-sを指定すると、指定フォルダに含まれるサブフォルダも計測対象となります。

4.操作手順
4.1.バッチ処理
4.1.3.計測対象一覧による計測方法

概要
コマンドライン引数の指定で、計測対象一覧ファイルに指定されるJavaファイルと、Javaファイルに定義されるClassファイルを、バッチ処理で計測をおこないます。

操作手順
1. コマンドライン引数に以下の指定をおこない、当ツールを起動します。

   -b -l [計測対象一覧ファイル名]

   実行例：
   c:¥stj¥test.stjを対象の計測対象一覧ファイルとした場合
   java com.fujitsu.simplia.StepCounter.metrics -b -l c:¥stj¥test.stj

※ 計測対象一覧ファイルとは、画面処理で指定した対象資産名／クラスパス／オプションを格納した保存ファイルを指します。

4.操作手順
　4.1.バッチ処理
　　4.1.4.計測単位の指定方法

概要

コマンドライン引数の指定で、計測単位を指定し、バッチ処理で計測をおこないます。

操作手順

1. クラス単位で計測をおこなう場合、コマンドライン引数に以下の指定をおこない、当ツールを起動します。計測単位を指定しない場合、クラス単位に計測がおこなわれます。

   -b [対象資産パラメタ]

   実行例：
   c:¥Java¥test.javaを計測対象のJavaファイルとした場合
   java com.fujitsu.simplia.StepCounter.metrics -b c:¥Java¥test.java

2. メソッド単位に計測をおこなう場合、コマンドライン引数に以下の指定をおこない、当ツールを起動します。

   -b [対象資産パラメタ]　-m

   実行例：
   c:¥Java¥test.javaを計測対象のJavaファイルとした場合
   java com.fujitsu.simplia.StepCounter.metrics -b c:¥Java¥test.java -m

※ 対象資産パラメタは、ファイル指定／フォルダ指定を指します。対象資産パラメタに計測対象一覧ファイルが指定される場合、計測単位の指定はできません。この場合、計測対象一覧ファイルに指定される計測単位が有効になります。

4.操作手順
　4.1.バッチ処理
　　4.1.5.計測結果出力先の指定方法

概要

コマンドライン引数の指定で、計測結果の出力先を指定し、バッチ処理で計測をおこないます。

操作手順

1. コマンドライン引数に以下の指定をおこない、当ツールを起動します。

   -b [対象資産パラメタ] -csv [ＣＳＶファイル名]

   実行例：
   c:¥Java¥test.javaを計測対象のJavaファイルとし、 c:¥csv¥test.csvを出力先のＣＳＶファイルとした場合
   java com.fujitsu.simplia.StepCounter.metrics -b c:¥Java¥test.java -csv c:¥csv¥test.csv

※ 対象資産パラメタは、ファイル指定／フォルダ指定を指します。対象資産パラメタに計測対象一覧ファイルが指定される場合、出力先の指定はできません。この場合、計測対象一覧ファイルに指定される出力先が有効になります。
※ CSVファイル名には、書き込み可能なファイル名を指定します。
※ -csvパラメタが指定されない場合、計測結果は標準出力に出力されます。

4.操作手順

4.2.画面処理

4.2.1.新規計測対象一覧ファイル指定方法

概要

新規に計測する計測対象への各種設定を行います。

操作手順

1. メイン画面の「ファイル」メニューの「新規作成」コマンドを選択します。

2. 各種設定（※）を設定します。

3. メイン画面の「ファイル」メニューの「名前を付けて保存」コマンドを選択し、計測対象一覧ファイルを保存します。

※設定内容は以下のとおりです。

設定画面	設定項目	内容
メイン画面	ＣＳＶファイル	計測結果の出力先ファイル名を指定します。
	対象資産形式	計測対象の形式を選択します。
	対象資産名	計測対象となるファイルまたはフォルダを指定します。
クラスパス指定ダイアログ	クラスパス名	Classファイルの検索パスを指定します。 (クラス単位の計測をおこなう場合、Classファイルが存在するパス名が環境変数CLASSPATHに設定されていない場合、設定します。メソッド単位の計測をおこなう場合、設定の必要はありません。)
オプション指定ダイアログ	計測単位	計測単位を選択します。 (初期状態ではクラス単位が選択されています。)
アラーム指定ダイアログ	アラーム表示内容	計測結果を表示した場合のアラーム表示内容を設定します。 (初期状態ではパーセンタイル形式、閾値25％が設定されています。)

4.操作手順
　4.2.画面処理
　　4.2.2.対象資産の指定方法

概要

計測対象となる対象資産名を指定します。

操作手順

1. メイン画面の「対象資産形式」コンボボックスから計測対象となる資産の指定形式を選択します。

2. メイン画面の［対象資産名］横の［参照］ボタンの選択により、ファイル選択ダイアログボックスが表示されます。ファイル選択ダイアログボックスから対象資産名を選択します。対象資産は、１．で指定した対象資産形式の内容により、以下の形式で選択されます。

「対象資産形式」の選択内容	［対象資産名］横の［参照］ボタンでの選択内容
ファイル形式	計測対象となるJavaファイル名を選択します。ファイル選択ダイアログボックスでは、計測対象ファイルを直接選択します。
フォルダ形式 （サブフォルダを含まない）	計測対象となるフォルダ名を選択します。フォルダ選択ダイアログボックスでは、計測対象フォルダを直接選択します。
フォルダ形式 （サブフォルダを含む）	同上

3. メイン画面の「追加」ボタンを選択し、２．で選択した対象資産名を「対象資産一覧」リストに追加します。「対象資産一覧」リストに追加されない対象資産は、計測の対象とはなりません。

4.操作手順
　4.2.画面処理
　　4.2.3.計測対象一覧ファイルの読み込み方法

概要

作成済の計測対象一覧ファイルを読み込みます。

操作手順

1. メイン画面の「ファイル」メニューの「計測対象一覧フィルを開く」コマンドを選択します。

2. 表示されるファイル選択ダイアログボックスから、必要な計測対象一覧ファイルを選択します。

※ 旧ツールバージョン（V50L10）にて作成された計測対象一覧ファイルが選択された場合、「上限値／下限値」形式によるアラーム表示を行います。

4.操作手順
　4.2.画面処理
　　4.2.4.クラスパスの指定方法

概要

当ツールでは、クラス単位の計測時に対象資産として指定されるJavaファイルからClass名を特定し、Classファイルを検索し計測します。このため、Classファイルの検索パス（フォルダ名／Ｊａｒファイル名）をクラスパスとして指定します。

環境変数CLASSPATHが存在する場合、環境変数CLASSPATHへの指定パスも検索対象となるため、環境変数CLASSPATHの内容は、クラスパスとして指定する必要はありません。また、環境変数CLASSPATHに必要な検索パスがすべて設定されている場合や、メソッド単位の計測をおこなう場合は、当パラメタの指定は必要ありません。

Classファイルは、環境変数CLASSPATHへの指定順、クラスパスでの指定順で検索されます。

> 注意）Classファイルがフォルダに展開されている場合のフォルダ構成や、Jarファイルの内のディレクトリ構成は、通常のJava実行環境でのclasspath指定と同じく、パッケージと同じ構成になっている必要があります。

> 注意) ClassファイルまたはJarファイルへのフルパスの長さが、OSの制限以上になっている場合、参照できないためにエラーが発生することがあります。

操作手順

1. メイン画面の「オプション」メニューの「クラスパス指定」コマンドを選択します。

2. クラスパス指定ダイアログボックスで、クラスパスとなるフォルダ名・Jarファイル名の入力を行います。「ファイル参照」ボタン・「フォルダ参照」ボタンを選択することにより、ファイル選択ダイアログ・フォルダ選択ダイアログから選択することも可能です。

   <u>a.ファイル選択ダイアログ</u>
   クラスパスとなるJarファイル名を選択します。選択されたJarファイル名が「クラスパス名」テキストフィールドに設定されます。

   <u>b.フォルダ選択ダイアログ</u>
   クラスパスとなるフォルダ名を選択します。選択されたフォルダ名が「クラスパス名」テキストフィールドに設定されます。

   例1）クラス名がaaa.class、パッケージがcom.xxx.yyyの場合で、以下のフォルダ構成の場合
   　c:¥classes¥com¥xxx¥yyy¥aaa.class　→　フォルダ選択でc:¥classesを指定します

   例2）aaa.classがjarファイルに圧縮されて以下のように配置されている場合
   　c:¥classes¥jar¥abc.jar　→　ファイル選択でc:¥classes¥jar¥abc.jarを指定します

   例3）aaa.classがcom.xxx.bbbクラスを継承している場合
   　c:¥classlib¥com¥xxx¥bbb.class　→　フォルダ選択でc:¥classlibを追加指定します

3. クラスパス指定ダイアログボックスで、「追加」ボタンを選択し、２．で選択したクラスパス名を、「クラスパス一覧」リストに追加します。「クラスパス一覧」リストに追加されない場合、検索パスとして設定されません。

4. 必要なクラスパスを設定後、クラスパス指定ダイアログボックスの「ＯＫ」ボタンを選択し、指定内容を保存し、ダイアログボックスを終了します。

4.操作手順
　4.2.画面処理
　　4.2.5.計測単位の指定方法

概要

指定対象資産の計測単位を指定します。

操作手順

1. メイン画面の「オプション」メニューの「計測オプション」コマンドを選択します。

2. 表示されるオプション指定ダイアログボックスから、計測単位(クラス単位／メソッド単位)を選択します。

3. オプション指定ダイアログボックスの「ＯＫ」ボタンを選択し、指定内容を保存しダイアログボックスを終了します。

概要

計測結果表示画面で表示される計測値のアラーム表示指定をおこないます。

操作手順

1. メイン画面の「オプション」メニューの「アラーム指定」コマンドまたは、計測結果表示画面の「オプション」メニューの「アラーム指定」コマンドを選択します。

2. 「パーセンタイル」ラジオボタン・「上限値／下限値」ラジオボタンより、アラーム表示形式を選択します。

3. アラーム表示の対象項目を設定します。

   a.パーセンタイル選択時
   すべての計測項目がアラーム表示対象となります。対象項目の設定はありません。

   b.上限値／下限値選択時
   アラーム指定ダイアログボックスの「計測項目」コンボボックスから、アラーム指定の対象となる計測項目を選択します。

4. アラーム範囲を設定します。

   a.パーセンタイル選択時
   アラーム指定ダイアログボックスの「閾値」のテキストフィールドに、アラーム範囲（下限の閾値※）を設定します。計測結果表示画面でのアラーム表示は、指定された閾値より項目毎のパーセント点（上限／下限）を算出し、パーセント点（上限）以上の計測値、および、パーセント点（下限）以下の計測値に対しておこなわれます。詳細については、「7.3.パーセンタイル形式のアラーム表示」を参照してください。

   b.上限値／下限値選択時
   アラーム指定ダイアログボックスの「上限値」「下限値」のテキストフィールドに、アラーム範囲を設定します。計測結果表示画面でのアラーム表示は、上限値以上の計測値、または下限値以下の計測値に対しておこなわれます。

5. アラーム指定ダイアログボックスの「ＯＫ」ボタンを選択し、指定内容を保存しダイアログボックスを終了します。

※下限の閾値は、０％～５０％の範囲内で設定します。

4.操作手順
　4.2.画面処理
　　4.2.7.計測結果の表示方法

概要

対象資産の計測結果を表示します。

操作手順

計測と計測結果表示

1. メイン画面の「計測」メニューの「計測」コマンドを選択し、対象資産の計測をおこないます。

2. 計測終了後に、計測結果表示画面が起動され、計測結果が表示されます。

計測済結果表示

1. メイン画面の「表示」メニューの「計測結果表示」コマンドを選択し、計測結果表示画面を表示します。

※ ただし、当操作は、当ツール起動後計測処理を実行した場合に有効です。当ツールの前回起動時の計測済結果を表示することはできません。
また、当操作では、直前の計測処理の結果のみ表示可能です。

4.操作手順
　4.2.画面処理
　　4.2.8.計測対象一覧ファイルの保存方法

概要

計測対象一覧ファイルを保存します。

操作手順

新規作成及びファイル名変更

1. メイン画面の「ファイル」メニューの「名前を付けて保存」コマンドを選択します。

2. 表示されるファイル選択ダイアログボックスから、保存先ファイル名を指定し「ＯＫ」ボタンを選択します。

上書き保存

1. メイン画面の「ファイル」メニューの「上書き保存」コマンドを選択します。

4.操作手順
　4.2.画面処理
　　4.2.9.計測方法

概要

指定の対象資産の計測をおこないます。

操作手順

1. メイン画面の「計測」メニューの「計測」コマンドを選択します。

　計測結果は、メイン画面で計測前に指定するＣＳＶファイルに出力されます。

5.計測内容説明

5.1.クラス単位の計測

概要

クラス単位の計測を実行した場合、指定の対象資産に含まれるJavaファイルをクラス単位に計測します。Javaファイル計測後、Javaファイルに定義されるクラス名をもとにClassファイルの計測をおこないます。Classファイルは、環境変数CLASSPATHに設定される検索パスと、当ツール内にクラスパス指定された検索パスから自動的に検出をおこない、計測されます。

> 注意）Classファイルがフォルダに展開されている場合のフォルダ構成や、jarファイルの内のディレクトリ構成は、通常のJava実行環境でのclasspath指定と同じく、パッケージと同じ構成になっている必要があります。

計測対象のクラスに、利用者独自またはシステムのクラスライブラリ（例えばservlet.jarなど）が含むクラスを使用／継承している場合、そのパッケージが存在するフォルダ名／Jarファイル名についても、環境変数CLASSPATHか当ツール内のクラスパス指定に検索パスとして設定されている必要があります。

また、ClassファイルはJavaファイルと同期が取れていて、最新状態になっていなければ、正常に計測をおこなうことができません。クラス単位の計測をおこなう前には、Javaファイルがコンパイル済であることを確認してください。

出力項目説明

出力項目	説明
クラス名	計測単位となるクラス名です。「パッケージ名．クラス名」の形式で出力されます。Javaファイルへの定義順で出力されます。Javaファイルに内部クラス・匿名クラスが定義される場合、以下の規則によりクラス名を生成します。 規則１） クラス直下に定義される内部クラスは、<u>定義元クラス$内部クラス</u>でクラス名を生成します。 規則２） メソッド内に定義される内部クラスは、<u>定義元クラス$連番$内部クラス</u>でクラス名を生成します。 規則３） 匿名クラスは、<u>定義元クラス$連番</u>でクラス名を生成します。 ※連番は、出現順を指し命名後のクラス名の重複を避けるための連番とします。
文数	クラス内の文数です。
継承数	クラスの継承数です。
フィールド変数の数	クラス内に定義されるフィールド変数の数です。
インスタンス変数の数	クラス内に定義されるインスタンス変数の数です。
パブリック変数の数	クラス内に定義されるパブリック変数の数です。
メソッド数	クラス内に定義されるメソッドの数です。

インスタンスメソッド数	クラス内に定義されるインスタンスメソッドの数です。
パブリックメソッド数	クラス内に定義されるパブリックメソッドの数です。
定義ファイル名	クラスが定義されるJavaファイル名です。Javaファイルの先頭に定義されるクラス名にのみ出力されます。

計測基準

計測項目	説明	計測ファイル
文数	実行文の数です。行数ではありません。また、コメント行／package文／import文／クラス定義文／メソッド定義文／変数定義文は、含まれません。	Java
継承数	上位クラスを継承する階層数です。	Class
フィールド変数の数	インスタンスとStatic両方のすべてを含む変数の数です。	Class
インスタンス変数の数	Static修飾子のない変数の数です。	Class
パブリック変数の数	public修飾子される変数の数です。	Class
メソッド数	全メソッド数です。	Class
インスタンスメソッド数	Static修飾子のないメソッド数です。	Class
パブリックメソッド数	public修飾子されるメソッド数です。	Class

Classファイルの計測に失敗した場合、計測項目には全角ハイフン（―）が出力されます。

計測基準の判定例を以下に示します。

```
Javaファイルの記述                        判定
----------------------------------------------------------------
package TEST.Package;
class TEST01 extends TEST00 {     →  TEST00が最上位クラスの場合、継承数は２
  int nDec;                       →  インスタンス変数／フィールド変数
  static public String str = "TEST01";→  パブリック変数／フィールド変数
  public TEST01() {
    nDec = 0;                     →  文数
  }
  int getDec() {                  →  インスタンスメソッド数
                                     ／メソッド数
    return nDec;                  →  文数
  }
  static public void printStr() { →  パブリックメソッド数
                                     ／メソッド数
    System.out.println(str);      →  文数
  }
  private String toDecString(){   →  インスタンスメソッド数
                                     ／メソッド数
    return Integer(nDec).toString(); →  文数
  }
};
```

5.計測内容説明
　5.2.メソッド単位の計測

概要

メソッド単位の計測を実行した場合、指定の対象資産に含まれるJavaファイルをメソッド単位に計測します。当計測では、Classファイルの計測はおこなわれません。

出力項目説明

出力項目	説明
メソッド名	計測単位となるメソッド名です。「パッケージ名．クラス名．メソッド名（メソッドの引数）」の形式で出力されます。
文数	メソッド内の文数です。この計測項目は、行数を指すものではありません。
分岐文数	メソッド内の条件分岐文の数です。if文（else if文）／switch文内のcase文の数です。単独のelse文とswitch文内のdefault文は、対象外とします。
ループ数	メソッド内のループ文の数です。for文／while文／do～while文の数です。
メソッドＣＡＬＬ	他メソッドの実行数です。
定義ファイル名	メソッドが定義される計測対象となったJavaファイル名です。Javaファイルの先頭に定義されるメソッド名にのみ出力されます。

計測基準

計測項目	説明	計測ファイル
文数	実行文の数です。行数ではありません。また、コメント行、変数定義文は、含まれません。	Java
分岐文数	if文（else if文）／switch文内のcase文の数です。単独のelse文とswitch文内のdefault文は、対象外とします。	Java
ループ数	for文／while文／do～while文の数です。	Java
メソッドＣＡＬＬ数	他メソッドの実行数です。	Java

計測基準の判定例を以下に示します。

Javaファイルの記述　　　　　　　　　　判定

```
package TEST.Package;

class TEST01 extends TEST00 {
  int nDec;
  static public String str = "TEST01";

  static public void printStr() {
    String strOut = "";              →文数
    for(int i=0;i<10;i++) {          →ループ数／文数
      if( i==0 ) {                   →分岐文数／文数
```

```
        strOut += "CLASS";                →文数
      } else {
        strOut += str;                    →文数
      }
    }
    print(strOut);                        →メソッドCALL数／文数
  }
  static private void print(String str) {
    System.out.println(str);
  }
};
```

6.計測内容説明

6.1.ＣＳＶファイル

概要

計測内容は、ＣＳＶファイルに出力されます。画面起動時の計測で、計測処理を中断ボタンにより中断した場合、計測された部分までの内容が出力されます。ＣＳＶファイルは、計測単位により出力内容が異なります。

出力内容説明

（１）クラス単位計測時

"class-name",999,999,999,999,999,999,999,999,"file-name"
　　a　　　b　c　d　e　f　g　h　i　　j

項	計測項目	説明
a	クラス名	計測対象となるクラス名を「パッケージ名．クラス名」の形式で出力されます。
b	文数	クラス内の文数が出力されます。
c	継承数	クラスの継承数が出力されます。
d	フィールド変数の数	クラス内に定義するフィールド変数の数が出力されます。
e	インスタンス変数の数	クラス内に定義するインスタンス変数の数が出力されます。
f	パブリック変数の数	クラス内に定義するパブリック変数の数が出力されます。
g	メソッド数	クラス内に定義するメソッド数が出力されます。
h	インスタンスメソッド数	クラス内に定義するインスタンスメソッド数が出力されます。
i	パブリックメソッド数	クラス内に定義するパブリックメソッド数が出力されます。
j	ファイル名	クラスが定義されるJavaファイル名をフルパスで出力されます。 Javaファイル内で先頭に定義されるクラスにのみ出力されます。

（２）メソッド単位計測時

"method-name",999,999,999,999,"file-name"
　　a　　　b　c　d　e　　f

項	計測項目	説明
a	メソッド名	計測対象となるメソッド名を「パッケージ名．クラス名．メソッド名（引数）」の形式で出力されます。
b	文数	メソッド内の文数が出力されます。
c	分岐文数	メソッド内の条件分岐文の数が出力されます。
d	ループ数	メソッド内のループ文の数が出力されます。
e	メソッドCALL数	メソッド内での他メソッドのCALL数が出力されます。
f	ファイル名	メソッドが定義されるJavaファイル名をフルパスで出力されます。Javaファイル内で先頭に定義されるメソッドにのみ出力されます。

7.アラーム指定
7.1.測定値の判定

概要
当ツールでの計測内容は、クラスやメソッドの文数により目標値と実際の開発量から、開発進捗状況を把握することができます。その他に、クラスやメソッドの設計品質の判定に活用することができます。判定内容を以下に示します。

判定内容
（１）メソッド数が大き過ぎる場合
メソッド数が大き過ぎる場合、クラスに機能が詰め込まれ過ぎている恐れがあります。全体としてメソッド数が大きい場合は、内容の理解が難しくなり、保守や再利用がしづらくなる可能性もあります。この場合、クラスの単位の考え方を見直し、適切な単位に細分できないか検討する必要があります。また、一部に極端に大きなクラスやクラス群(パッケージ)があって平均値を大きくしている場合は、他の部分との機能分担を見直す必要があると考えられます。

（２）変数の数が大き過ぎる場合
基本的にはメソッド数の場合と同様の問題があります。但し、定数フィールドについては、多少大きくても問題にする必要はありません。また、画面やDBエンティティなどのクラスは、他の値よりもかなり大きくなります。

（３）文数が大き過ぎる場合
上記同様、クラスに機能が詰め込まれ過ぎている可能性があります。特に、メソッド数に比較して、文数が大きいようであれば、手続き中心のプログラムになっている可能性が高いと考えられます。この場合、オブジェクトの持つ機能として、メソッドが適切に分割されているか考え直す必要があります。特定の大きな処理のみをメソッドとしていたのでは、部品としては使いづらく再利用性が低くなります。このような傾向が全体に見られる場合は、大きなレベルの設計/モデリングから見直す必要があります。

（４）メソッド数・フィールド変数の数・文数が小さ過ぎる場合
クラスが小さ過ぎれば、部品としての価値は低くなり、再利用性は低下します。また、分割によるオーバヘッドが大きくなり、実行性能の点でも好ましくありません。

（５）継承数が深過ぎる
継承の使い過ぎは、プログラムの可読性を低下させ、保守を困難にします。適度な深さになるように設計を見直すべきです。継承を多用した場合、上位のクラスの変更が困難になることもあり、委譲などによって機能を分割することも考える必要があります。但し、継承階層の深いシステムクラスを継承して用いている場合は、ユーザ定義の階層部分を判定対象とします。

（６）継承がほとんど使われていない
継承がほとんど使われていない場合、開発者がオブジェクト指向の考えを理解できていない可能性を示しています。クラスサイズの情報と合わせて、機能をサブクラス化できるような箇所がないか検討する必要があります。

7.アラーム指定
　7.2.アラーム指定基準値設定例

概要

計測結果表示画面でのアラーム指定基準値の設定例を、ビジネスソフトの事例をもとに以下に示します。

基準値

Javaプログラムのアラーム指定基準値の検討のために、以下のようなビジネスソフトの事例について測定をおこないました。

- A　ソフト開発発注の管理システム(A社殿)
  ファイル数 1132, 行数 約318K, クラス数 1177
- B　部品表システム(B社殿, クライアントのみ)
  ファイル数 709, 行数 約59K, クラス数 799
  これはフェーズ1とフェーズ2に分かれている。
- C　営業支援システム(C社殿)
  ファイル数 132, 行数 約36K, クラス数 186
- D　システム(D社殿、クライアントのみ)
  ファイル数 44, 行数 約8K, クラス数 44
- E　製造業向け設計業務 部門間ワークフロー(E社殿)
  ファイル数 136, 行数 約12K, クラス数 161

その結果を、事務処理ソフトの場合に大きな特徴を示すと思われる画面(Panel)クラス、エンティティクラス(DBテーブルに相当)と、それぞれ他のクラスと分類し集計しました。分類は、パッケージ及びクラスの名称をもとにしておこないました。一般には, 分布の特徴を見るには標準偏差がよく用いられますが, 当ツールの測定で得られるデータは、通常は正規分布でないため、有効ではありません。ここでは、四分位偏差をもとに分布の性質を調べ、基準平均値と上限値を設定しました。Ｊａｖａの基準値を下表に示します。

Javaの基準値

項目	画面(平均,上限)	ＤＢエンティティ(平均,上限)	一般(平均,上限)
フィールド数	15 - 50,100	20,80	3 - 10,20
メソッド数	10 - 35,80	40,160	5 - 10,30
文の数	100 - 400,700	100,300	25 - 80,300
非private非定数フィールド数	0	0	0

　平均値
一定の単位の平均値の基準であり、プログラム全体やパッケージ単位の特徴を調べる場合に用います。これを大きく外れていれば、その部分の設計に問題がある可能性があります。
　上限値
クラスや関数が異常であるかどうかを判定するための目安です。これを越えるクラスや関数は、構造に問題があると考えられるため、見直した方が良いでしょう。 なお、これらの基準は、おおよその目安として設定したものであり、絶対的な基準ではありません。実際の適用には、プロジェクトの必要に応じて、基準を設定することが望ましいでしょう。 例えば、通常のアプリケーションを設

計する場合と再利用を強く意識し汎用性を高めたライブラリを設計する場合とでは、基準は当然異なってきます。上記の基準は、通常のアプリケーションを意識したものです。再利用ライブラリの場合は、メソッド数等がより大きくなる傾向があります。

以下に分類毎の特徴を述べます。

画面クラスは、全体にサイズが大きくなる傾向があります。画面設計上の要件や使用するGUIビルダが生成するコードの特徴など種々の制約はありますが、プログラムの理解性を保つためには、上記の基準程度にとどめておくとよいでしょう。

DBエンティティクラスは、DBテーブルのサイズをもとにするので、フィールド数やメソッド数は大きくなります。通常は、データアクセスのメソッドが中心になるので、文の数は比較的小さくできます。これについて、基準は暫定値であり、はっきりしたことが言えない点がまだ多くあります。

その他のクラスについての基準値が、一般的なJavaのクラス設計基準と考えています。

参考として、C++での基準値を以下に示します。

C++の基準値

項目	基準平均	上限
データメンバの数(=フィールド数)	2 - 4	10
メソッド関数の数(=メソッド数)	3 - 13	30
文の数	20 - 120	500

分類毎のフィールド数／メソッド数／文数の集計結果を以下に示します。

（１）画面
画面クラスについては、比較的規模の大きいA、Bの結果を主に採用し、基準値を設定しました。

A(クラス数 131,インタフェース数 1)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	6503	49.27	253	0	59.90	29.0	57	235
メソッド数	3413	25.86	120	2	21.45	19.5	23	104
文数	62749	475.37	2897	0	521.22	307.0	609	2534

B(Phase1)(クラス数 50,インタフェース数 0)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	715	14.30	46	0	14.90	11.5	26	104
メソッド数	524	10.48	24	1	6.69	11.0	12	52
文の数	4958	99.16	309	3	89.21	78.5	146	600

B(Phase2)(クラス数 7,インタフェース数 0)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	197	28.14	48	14	11.95	29.0	22	103
メソッド数	157	22.43	33	16	7.61	17.0	15	76
文の数	2034	290.57	463	162	97.00	282.0	119	710

（２）ＤＢエンティティ
データベースエンティティについては、方式によって様々なバラツキがありました。データベースエンティティは、AのXXXEntityImplという名をもつクラスの値をベースに基準値を設定しました。

A(クラス数 275)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	5657	20.57	164	1	26.86	10.0	18	78
メソッド数	12695	46.16	334	4	53.58	24.0	35	157
文の数	24723	89.90	682	5	110.94	48.0	64	286

（３）その他
その他一般のクラスについては、これらの事例に加えて、Java標準ライブラリの測定結果、他の2つのミドルウェアソフトでの測定結果、及びC++について定めた値を参考にして基準を設定しました。

A(クラス数 238,インタフェース数 11)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	2169	7.66	254	0	21.10	3.0	6	25
メソッド数	2543	8.99	78	0	9.78	7.0	7	31
文の数	17558	62.04	996	0	103.80	33.0	39	173

B(Phase1)(クラス数 646,インタフェース数 23)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	4301	6.66	145	0	11.05	3.0	7	29
メソッド数	6503	10.07	138	0	11.82	7.0	10	43
文の数	50359	77.96	1319	0	143.67	33.5	72	300

B(Phase2)(クラス数 96,インタフェース数 0)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	699	7.28	99	0	13.54	2.0	8	33
メソッド数	843	8.78	116	0	16.00	5.0	9	37
文の数	7863	81.91	1496	0	181.70	31.0	84	340

C(クラス数 74,インタフェース数 0)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	243	3.28	66	0	9.12	1.0	0	1
メソッド数	507	6.85	137	0	18.60	3.0	2	10
文の数	1958	26.46	469	0	79.37	3.0	2	11

D(クラス数 30,インタフェース数 3)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	295	9.83	64	0	14.24	4.0	12	48
メソッド数	190	6.33	31	0	7.20	4.0	7	29
文の数	2402	80.07	398	0	97.57	49.0	123	500

E(クラス数 125,インタフェース数 8)

測定項目	合計値	平均値	最大値	最小値	標準偏差	中央値	四分位偏差	飛び離れ値
フィールド数	802	6.42	53	0	10.15	3.0	6	25
メソッド数	562	4.50	18	0	3.63	4.0	4	18
文の数	3352	26.82	145	0	31.43	18.0	35	144

7.アラーム指定
　7.3.パーセンタイル形式のアラーム表示

概要

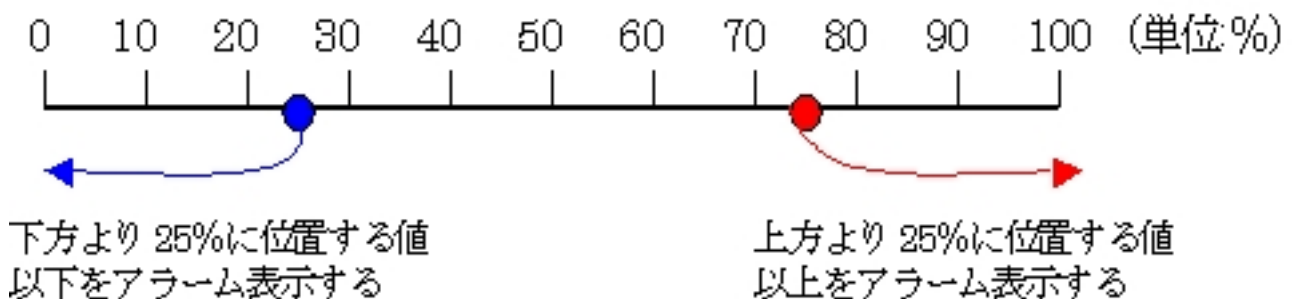
計測結果表示画面で表示される計測値について、パーセンタイル形式によるアラーム表示を行います。

表示説明

パーセンタイルは、計測したデータの分布を大きさの順に並べて、何パーセント目の値がどれくらいかを評価する統計的な表示法のひとつです。

パーセンタイル形式によるアラーム表示では、アラーム指定ダイアログボックスで指定された下限のパーセント点より、計測クラスのｐパーセンタイル（下限）／１００－ｐパーセンタイル（上限）を算出し、下限以下／上限を超える値について、アラーム表示を行います。

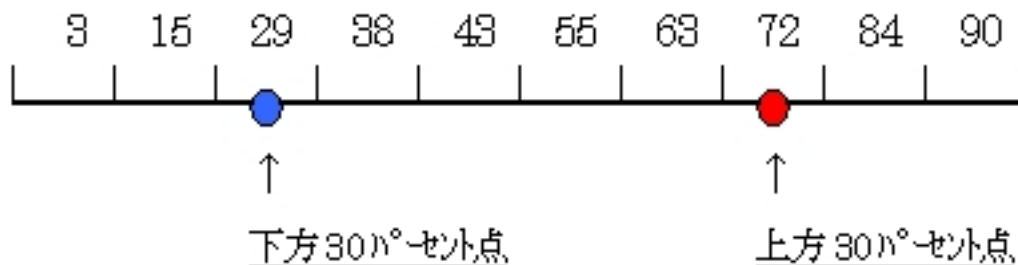
以下にパーセンタイルによるアラーム表示範囲の例を示します。

アラーム表示範囲の例）２５パーセンタイル

以下にパーセンタイルによるアラーム表示の例を示します。

例）クラスが１０個あって、それぞれの文数が以下のようだったとします。パーセント点＝３０（下限）指定時のパーセンタイル（上限／下限）を求め、アラームを表示します。

［各クラスの文数］15,63,84,29,43,55,90,3,72,38

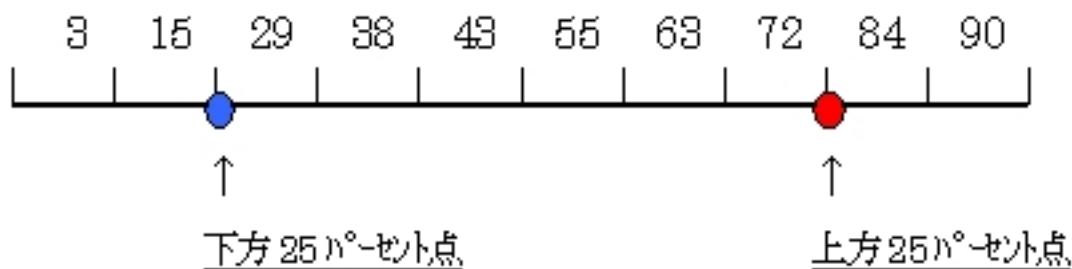
・上記データを大きさの順に並べ替えます
　3,15,29,38,43,55,63,72,84,90

・パーセンタイル（上限／下限）を求めます
　下限は、１０クラス中の３番目　…　文数＝２９
　上限は、１０クラス中の８番目　…　文数＝７２

・アラームを表示します

文数が２９（下限）以下を青色でアラーム表示します
文数が７２（上限）以上を赤色でアラーム表示します

［各クラスの文数］15,63,84,29,43,55,90,3,72,38

例）上記例にて、パーセント点＝２５（下限）指定時のパーセンタイル（上限／下限）を求め、アラームを表示します。

・パーセンタイル（上限／下限）を求めます
下限は、１０クラス中の２番目（１５）と３番目（２９）の中間　…　文数＝２２
上限は、１０クラス中の８番目（７２）と９番目（８４）の中間　…　文数＝７８

・アラームを表示します

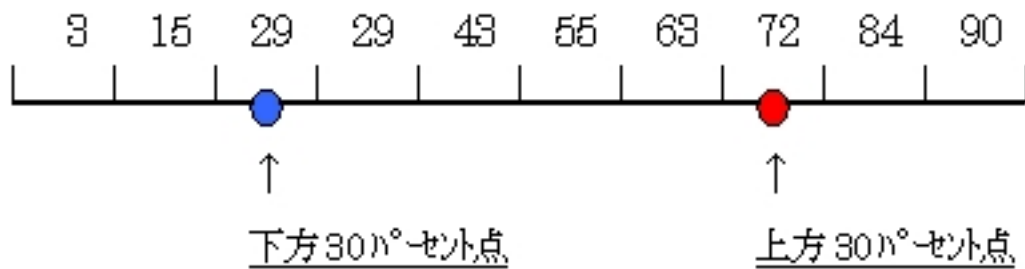
文数が２２（下限）以下を青色でアラーム表示します
文数が７８（上限）以上を赤色でアラーム表示します

［各クラスの文数］15,63,84,29,43,55,90,3,72,38

例）重複するデータが存在する場合のパーセント点＝３０（下限）を求め、アラームを表示します。

［各クラスの文数］3,15,29,29,43,55,63,72,84,90

・上記の重複する29をそれぞれカウントし、全件で１０件として割合を求めます

・パーセンタイル（上限／下限）を求めます
下限は、１０クラス中の３番目　…　文数＝２９
上限は、１０クラス中の８番目　…　文数＝７２

・アラームを表示します

文数が２９（下限）以下を青色でアラーム表示します
文数が７２（上限）以上を赤色でアラーム表示します

［各クラスの文数］3,15,29,29,43,55,63,72,84,90

8.制限・注意事項

8.1.注意事項

1.同一クラス名を定義するJavaファイルの計測について

計測対象として、同一クラス名（パッケージ名を含む）を定義する複数のJavaファイルを一度に指定する場合、計測するJavaファイルに定義されるクラスを正常に計測できません。Classファイルを環境変数CLASSPATHに指定されるパス／当ツール内のクラスパス指定で指定するパスの順で検索しているため、常に同一のClassファイルが検出されます。
同一クラス名（パッケージ名を含む）を定義する複数のJavaファイルを計測する場合は、計測処理を複数回に分割し実行する必要があります。

2.ユーザ定義のパッケージを使用するClassファイルの計測について

ユーザ定義のパッケージ内のクラスを使用するClassファイルの計測をおこなう場合、事前に環境変数CLASSPATHに、使用するパッケージが存在するパス名（フォルダ／Jarファイル）を指定する必要があります。指定されない場合、ユーザ定義のパッケージに含まれるクラスを使用するClassファイルを正常に計測できません。

3.ClassファイルとJavaファイルの同期について

クラス単位の計測では、ClassファイルはJavaファイルと同期が取れていて最新状態になっている必要があります。最新状態になっていない場合、正常に計測することができません。計測前にJavaファイルがコンパイル済であることを確認してください。

4.ＣＳＶファイルの出力について

当ツールでは、計測処理を実行した場合、指定したＣＳＶファイルの内容が計測結果で上書きされます。指定したＣＳＶファイルの既存情報が必要な場合は、出力先となるＣＳＶファイル名を変更するか、ＣＳＶファイルのバックアップを作成後、計測処理を実行してください。
コマンドライン引数に計測対象一覧ファイルを指定し、バッチ処理で計測処理を実行する場合、計測対象一覧ファイル内に指定されるＣＳＶファイル名に計測結果が出力されます。この場合も、指定のＣＳＶファイルに既存情報がある場合は、計測結果で上書きされるため、注意が必要です。

5.ヘルプ表示フレームの動作について

ヘルプ表示フレームでは、リンク先を選択するとＨＴＭＬページロードのため一時的に選択不可状態になります。選択不可状態が解除後にもＨＴＭＬページロードが継続している時に、頻繁にリンクが発生するとJavax.swingの例外メッセージがコマンドプロンプト上に表示されますことがありますが、処理上問題はありません。

6.JDKの動作環境について

当ツールの動作には、事前にJDKのbinフォルダが環境変数pathに追加されている必要があります。

7.ファイル選択・フォルダ選択時のネットワークドライブについて

ＣＳＶファイルの選択時、対象資産ファイル／対象資産フォルダの選択時、クラスパスファイル／クラスパスフォルダ選択時に起動されるファイル選択ダイアログ／フォルダ選択ダイアログでは、

直接ネットワークコンピュータをアクセスすることができません。ネットワークコンピュータをアクセスする際は、あらかじめ、エクスプローラ等にてネットワークドライブを割り当てておく必要があります。

8.制限・注意事項
　8.2.制限事項

1.計測対象となるJavaファイル／Classファイルについて

対象となるJavaファイルは、正常コンパイル可能なファイルで、シフトJISで記述されたファイルでなければなりません。また、対象となるClassファイルは、JDK1.3、1.4または5.0で作成されたファイルでなければなりません。

2.計測処理の中断について

画面処理で計測処理を中断する場合、中断時点まで計測した内容を計測結果として出力します。
バッチ処理や画面処理起動時に、コマンドプロンプトからアプリケーションを中断する場合、計測結果は出力されません。

3.複数起動について

当ツールは、複数起動をサポートしていません。このため、複数起動時の動作は保証されません。

4.起動時のフォルダについて

当ツールを起動する場合、ツールのインストールフォルダがカレントフォルダになっていなければ、動作は保証されません。

5.ツールの構成について

当ツールがインストールされる、インストールフォルダには、Ｔｅｍｐ／Ｈｅｌｐサブフォルダが存在します。インストールフォルダ内のこれらのサブフォルダを変更・削除すると、当ツールの動作は保証されません。

6.SQLJ使用ソースの動作制限

当ツールでは、SQLJを使用したJavaソースの計測を対応していません。SQLJを使用したJavaソースを計測対象とした場合、そのソースは計測されません。

7.パス名長の制限

当ツールでは、使用するファイルを260バイトまで指定可能です。ファイルシステムによっては、260バイトまでの指定ができません。この場合、使用可能なバイト数分のみパス名として採用され、使用可能バイトを超える部分は切り捨てられます。最大長のパス名を指定する場合は、注意が必要です。

8.JDK5.0版の制限

JDK5.0版では以下の制限があります。

●列挙型について

- enumクラスを宣言した場合、valueOf(),values()という２つのpublic staticなメソッドが追加

されるため、記述したメソッド数より+2されます。

- enumクラスを宣言した場合で、メソッドを宣言している場合、内部的なメソッドが保持されるためさらに+1されます。
- 通常のクラス等でenum型フィールドを宣言した場合でも、クラスとして別に計測されます。

●注釈について

- ステップ数計測機能では記述された量を計測するという観点から計測対象としますが、注釈行(コメント）ではなく有効行として計測します。一方、本メトリクス機能ではソースの複雑度／特性を見るという観点から、文数の計測対象とはなりません。
- 注釈型を定義したソースをメソッド計測した場合、処理が存在しないため計測結果が出力されません。
- クラス／インタフェース宣言内で注釈型を定義したソースは計測できません。

9.エラーメッセージ
　9.1.メッセージ一覧

メッセージの取得に失敗しました。

【意味】
メッセージの取得に失敗しました。

【対処】
当ツールが正常にインストールされていない可能性があります。インストール後再度実行してください。

STJ-001 当ツールの実行には、ＪＤＫ１．３以降が必要です。

【意味】
当ツールの実行には、ＪＤＫ１．３以降が必要です。

【対処】
ＪＤＫ１．３以降をインストール後、再度実行してください。

STJ-002 アラーム指定の閾値の内容が不当です。

【意味】
アラーム指定ダイアログでの設定で、アラーム範囲（閾値）の設定に誤りがあります。

【対処】
アラーム範囲の設定を修正します。

STJ-003 アラーム指定の上限値／下限値の内容が不当です。

【意味】
アラーム指定ダイアログでの設定で、アラーム範囲（上限値／下限値）の設定に誤りがあります。

【対処】
アラーム範囲の設定を修正します。

STJ-004 ＣＳＶファイル名が指定されていません。

【意味】
メイン画面でＣＳＶファイル名が指定されていません。

【対処】
ＣＳＶファイル名を指定します。

STJ-005 計測対象のクラスまたは継承元クラスを検出できません。

JAVA=Ｊａｖａファイル名
CLASS=クラス名

【意味】
Classファイルの計測時に、計測対象となるクラスまたは、継承元クラスを検出できません。

【対処】
計測対象となるクラスまたは、継承元クラスが存在するパスを、当ツールのクラスパス指定に設定します。

例1）クラス名がaaa.class、パッケージがcom.xxx.yyyの場合で、以下のフォルダ構成の場合
c:¥classes¥com¥xxx¥yyy¥aaa.class　→　フォルダ選択でc:¥classesを指定します

例2）aaa.classがjarファイルに圧縮されて以下のように配置されている場合
c:¥classes¥jar¥abc.jar　→　ファイル選択でc:¥classes¥jar¥abc.jarを指定します

例3）aaa.classがcom.xxx.bbbクラスを継承している場合
c:¥classlib¥com¥xxx¥bbb.class　→　フォルダ選択でc:¥classlibを追加指定します

注意）フォルダに展開されている場合のフォルダ構成や、Jarファイルの内のディレクトリ構成は、通常のJava実行環境でのclasspath指定と同じく、上記例1)のようにパッケージと同じ構成になっている必要があります。

　また、ClassファイルまたはJarファイルへのフルパスの長さが、OSの制限以上になっている場合、参照できないためにこのエラーが発生することがあります。

STJ-006 クラスパスの指定が不当です。

【意味】
クラスパス指定ダイアログボックスで、クラスパス名の指定内容に誤りがあります。

【対処】
クラスパス名への指定内容を修正します。

STJ-007 クラスパスへの指定が重複しています。

【意味】
クラスパス指定ダイアログボックスで、クラスパス名の指定内容が重複しています。

【対処】
クラスパス名に指定した内容が、すでにクラスパス一覧に設定されているため、設定する必要はありません。

STJ-008 計測処理で異常が発生しました。

【意味】
計測処理中に異常が発生しました。

【対処】
標準出力または、当ツールのインストールフォルダ配下のＴｅｍｐフォルダに出力されるエラーファイル「mfstpjm.err」に出力されるエラーメッセージを確認し、メッセージごとに対処します。

STJ-009 計測処理を中断します.
よろしいですか？

【意味】
計測中に、中断ダイアログボックスで中断ボタンが選択されました。

【対処】
計測処理を中断する場合は「はい」ボタンを、中断しない場合は「いいえ」ボタンを選択します。

STJ-010 計測処理が正常終了しました。

【意味】
計測処理が正常終了しました。

【対処】
なし。

STJ-011 指定ファイルを読み込むことができません。
　TYPE=ファイル種別
　FILE=ファイル名

【意味】
指定ファイルの読み込みに失敗しました。

【対処】
指定ファイルが、読み込み可能な指定ファイル種別のファイルであることを確認します。必要であれば、指定ファイル名を修正し再度実行します。

STJ-012 指定ファイルに書き込むことができません。
　TYPE=ファイル種別
　FILE=ファイル名

【意味】
指定ファイルの書き込みに失敗しました。

【対処】
指定ファイルが、書き込み可能な指定ファイル種別のファイルであることを確認します。必要であれば、指定ファイル名を修正し再度実行します。

STJ-013 計測結果を表示中です。
計測結果表示フレームを終了してください。

【意味】
計測結果表示画面を表示中のため、選択した処理を実行できません。

【対処】
計測結果表示画面を終了し、再度実行してください。

STJ-014 エラー情報の表示に失敗しました。
標準出力またはエラーファイル(mfstpjm.err)を参照してください。

【意味】
エラー情報表示画面の表示に失敗しました。

【対処】
エラー出力先を標準出力に設定し、再度計測するか、当ツールのインストールフォルダ配下のＴｅｍｐフォルダに出力されるエラーファイル「mfstpjm.err」の内容を確認します。

STJ-015 計測処理で異常が検出されました。
エラーファイルを表示しますか？

【意味】
計測処理で異常が発生しました。

【対処】
エラー情報表示画面によりエラーファイルの内容を表示する場合は「はい」ボタンを、表示しない場合は「いいえ」ボタンを選択します。

STJ-016 ＨＴＭＬヘルプの表示に失敗しました。
WWWブラウザから参照してください。

【意味】
ヘルプ表示画面の表示に失敗しました。

【対処】
ｗｗｗブラウザを使用し、ヘルプを参照します。

STJ-017 Ｊａｖａファイルの形式が正しくありません。
FILE=Javaファイル名
ERR=異常箇所

【意味】
指定のJavaファイルの形式が正しくありません。

【対処】
Javaファイル内の異常箇所を確認し、修正します。指定するJavaファイルは、正常コンパイルされるファイルでなければなりません。

STJ-018 対象資産に計測対象となるＪａｖａファイルが存在しません。
PATH=対象資産名

【意味】
指定の対象資産内に計測対象となるJavaファイルが存在しません。

【対処】
計測対象となるJavaファイルが存在する、対象資産名を指定します。

STJ-019 ファイルの読み込みに失敗しました。
TYPE=ファイル種別

FILE=ファイル名

【意味】
指定ファイルの読み込みに失敗しました。

【対処】
指定ファイル名が、ファイル種別に合った読み込み可能なファイルであることを確認します。

STJ-020 ファイルの書き込みに失敗しました。
TYPE=ファイル種別
FILE=ファイル名

【意味】
指定ファイルの書き込みに失敗しました。

【対処】
指定ファイル名が、ファイル種別に合った書き込み可能なファイルであることを確認します。

STJ-021 ファイルの形式が正しくありません。
TYPE=ファイル種別
FILE=ファイル名

【意味】
指定ファイルの形式が、指定ファイル種別の形式と一致しません。

【対処】
指定ファイル名が、ファイル種別に合った形式のファイルであることを確認します。

STJ-022 ファイルが存在しません
TYPE=ファイル種別
FILE=ファイル名

【意味】
指定ファイルが存在しません。

【対処】
ファイル種別に合った既存ファイル名を指定します。

STJ-023 フィールド情報を検出できません。
JAVA=Javaファイル名
CLASS=クラス名

【意味】
Classファイルの計測時に、フィールド情報の検出に失敗しました。

【対処】
計測対象となるクラス内にユーザが定義するパッケージが使用されているかどうか確認します。使用している場合は、ツール内のクラスパス指定により、使用しているパッケージが存在するパス（フォルダ／Jarファイル）名を指定します。指定されていない場合、計測対象となるClassファイルが正常ファイルであることを確認します。

STJ-024 メソッド情報を検出できません。
　JAVA=Javaファイル名
　CLASS=クラス名

【意味】
Classファイルの計測時に、メソッド情報の検出に失敗しました。

【対処】
計測対象となるクラス内に、ユーザが定義するパッケージが使用されているかどうか確認します。使用している場合は、ツール内のクラスパス指定により、使用しているパッケージが存在するパス（フォルダ／Jarファイル）名を指定します。指定されていない場合、計測対象となるClassファイルが正常ファイルであることを確認します。

STJ-025 対象資産名の指定が重複しています。

【意味】
メイン画面で、対象資産名の指定に重複があります。

【対処】
メイン画面で指定される対象資産名が、すでに対象資産一覧に存在するため、指定の必要がありません。対象資産形式がサブフォルダを含むフォルダの場合、上位フォルダを指定するよう修正します。

STJ-026 対象資産が存在しません。

【意味】
メイン画面で指定する対象資産が存在しません。

【対処】
既存の対象資産を指定します。

STJ-027 対象資産名が指定されていません。

【意味】
メイン画面で対象資産一覧に追加するための対象資産名が指定されていません。

【対処】
既存の対象資産名を指定し、追加処理を実行します。

STJ-028 対象資産形式が変更されました。
対象資産一覧への選択内容は、消去されます。
処理を続行しますか？

【意味】
メイン画面で対象資産形式が変更されました。対象資産一覧に指定される資産名は、形式が一致しないため、対象資産一覧から資産名を消去します。

【対処】
対象資産一覧から資産名を消去し、対象資産形式を変更する場合は「はい」ボタンを、対象資産一覧から資産名を消去しないで、対象資産形式を変更しない場合は「いいえ」ボタンを選択します。

STJ-029 対象資産の形式が不当です。

【意味】
メイン画面の対象資産名に指定する資産の形式が、対象資産形式で選択した形式と一致しません。

【対処】
対象資産形式の形式内容と一致する、対象資産名を指定します。

STJ-030 対象資産の形式が不当です。
　OBJECT=対象資産名

【意味】
メイン画面の対象資産名に指定する資産の形式が、対象資産形式で選択した形式と一致しません。

【対処】
対象資産形式の形式内容と一致する対象資産名を指定します。

STJ-031 コマンドライン引数の指定に誤りがあります。

【意味】
当ツール起動時に指定するコマンドライン引数の指定に誤りがあります。

【対処】
コマンドライン引数の内容を修正し、再度実行します。

STJ-032 計測対象一覧ファイルが保存されていません。
保存しますか？

【意味】
メイン画面／クラスパス指定ダイアログボックス／オプション指定ダイアログボックス／アラーム指定ダイアログボックスで指定された内容が保存されていません。

【対処】
指定内容を保存する場合は「はい」ボタンを、保存しない場合は「いいえ」ボタンを選択します。

STJ-033 計測対象のＪａｖａソースの更新日より古いクラスファイルです。
　JAVA=Javaファイル名
　CLASS=クラス名

【意味】
計測対象のJavaファイルの更新以前にClassファイルが作成されています。

【対処】
計測対象Javaファイルをコンパイルし、作成されたClassファイルと同期がとれていることを確認します。また、当ツールのクラスパス指定に設定して、Classファイルの検索パスが、正常であることを確認します。

STJ-034 エラーファイルの出力に失敗しました。
エラーメッセージは標準出力に出力されます。

【意味】

エラーファイル(mfstpjm.err)の出力に失敗しました。計測時のエラーメッセージは、標準出力に出力します。

【対処】
当ツールのインストールフォルダ配下のＴｅｍｐフォルダのエラーファイル(mfstpjm.err)が出力可能なファイルであることを確認してください。

STJ-035 JDKのbinフォルダの参照に失敗しました。

【意味】
JDKインストールフォルダのbinフォルダの参照に失敗しました。

【対処】
環境変数pathにJDKのbinフォルダを追加し再度実行します。

STJ-036 指定のパスは既に存在します。
上書きしますか？

【意味】
指定のパスは既に存在します。

【対処】
上書きする場合は「はい」ボタンを、上書きしない場合は「いいえ」ボタンを選択します。

STJ-037 指定のパスは存在しません。
既存のパスを指定してください。

【意味】
指定のパスは存在しません。

【対処】
既存のパスを再度指定します。

10.サンプル

10.1.サンプルの使い方

MF-STEPCOUNTERのインストールフォルダには"sample"フォルダが作成され、サンプルソースが格納されています。

ファイル名	説明
Sample2.java	サンプルのアプレットソース
SAMPLE2.jar	Sample2.javaから生成されるclassファイルを格納したjarファイル

10.1.1 サンプルソースを計測するには

1. Javaソフトウェアメトリクス計測機能をウィンドウズのスタートメニューから起動するか、起動されているJavaソフトウェアメトリクス計測機能の［ファイル］メニューから［新規作成］コマンドを実行します。
2. メイン画面の［CSVファイル］横の［参照］ボタンを押して、計測結果を出力するCSVファイル名を［ファイル選択］ダイアログから指定します。
3. メイン画面の［対象資産形式］のリストから、［ファイル指定］を選択します。
4. メイン画面の［対象資産］横の［参照］ボタンを押して、sampleフォルダにある"Sample2.java"ファイルを［ファイル選択］ダイアログから指定します。
5. メイン画面の［追加］ボタンを押して、［対照資産一覧］に追加します。
6. 計測対象となる"Sample2.java"に対応するclassファイルの場所を、次のどちらかの方法でクラスパスに登録します。
    - a. 環境変数のclasspathに追加する
    - b. ［オプション］メニューの［クラスパス指定］コマンドを実行して、［クラスパス指定ダイアログボックス］から追加する

   b.の場合、［クラスパス指定ダイアログボックス］の［ファイル参照］ボタンを押して、sampleフォルダにある"SAMPLE2.jar"ファイルを［ファイル選択］ダイアログから指定し、［追加］ボタンにより追加します。
   尚、計測対象資産から参照されるクラス(JDKなど)のクラスパスは、必ず環境変数のclasspathに追加しておいてください。
7. メイン画面の［計測］メニューから［計測］コマンドを実行します。
8. 計測結果は指定されたCSVファイルに保存され、計測結果が画面表示されます。

詳細については以下を参照してください。

- 新規計測対象一覧ファイル指定方法
- 対象資産の指定方法
- クラスパスの指定方法
- ＣＳＶファイル

10.1.2 計測方法を変更するには

初期設定では「クラス単位の計測」になっていますが、「メソッド単位の計測」に変更して、クラス内のより詳細な計測をすることができます。

1. メイン画面の［オプション］メニューから［計測オプション］コマンドを実行します。
2. ［オプション指定ダイアログボックス］で［メソッド単位の計測］を選択します。
3. メイン画面の［計測］メニューから［計測］コマンドを実行します。
4. 計測結果は指定されたCSVファイルに保存され、計測結果が画面表示されます。

詳細については以下を参照してください。

- クラス単位の計測
- メソッド単位の計測

10.1.3 アラーム表示するには
パーセンタイル形式の閾値指定、または、各種計測項目の上限値／下限値を指定することにより、計測結果表示画面でアラーム表示することができます。

- 上限値以上の値は赤色表示します
- 下限値以下の値は青色表示します

[パーセンタイル指定]

1. メイン画面の［オプション］メニューから［アラーム指定］コマンドを実行します。
2. ［アラーム指定ダイアログボックス］の「パーセンタイル」ラジオボタンを選択します。
3. 閾値を設定します。
4. メイン画面の［計測］メニューから［計測］コマンドを実行して再度計測するか、［表示］メニューから［計測結果表示］コマンドにより計測結果を再表示します。
5. 計測結果表示画面の［オプション］メニューの［アラーム指定］コマンドからも指定できます。

[上限値／下限値指定]

1. メイン画面の［オプション］メニューから［アラーム指定］コマンドを実行します。
2. ［アラーム指定ダイアログボックス］の「上限値／下限値」ラジオボタンを選択します。
3. ［項目］リストから任意の項目を選択します。
4. 上限値と下限値を設定します。
5. メイン画面の［計測］メニューから［計測］コマンドを実行して再度計測するか、［表示］メニューから［計測結果表示］コマンドにより計測結果を再表示します。
6. 計測結果表示画面の［オプション］メニューの［アラーム指定］コマンドからも指定できます。

新規計測時は、「パーセンタイル指定」「閾値=２５％」がデフォルトで設定されます。
アラームを表示しない場合は、閾値、および、上限値／下限値に空白（データなし）を設定します。

10.1.4 計測対象を保存するには
指定した計測対象や出力先CSVファイル、各種オプション設定などを［計測対象一覧ファイル］として保存することができます。

1. メイン画面の［ファイル］メニューから［名前を付けて保存］コマンドを実行します。

2. ［ファイル名を付けて保存ダイアログボックス］で保存先を指定します。
3. ［ファイル名を付けて保存ダイアログボックス］の［保存］ボタンにより保存されます。

10.1.5 再計測するには

保存された［計測対象一覧ファイル］を読み込むことにより、簡単に再計測が行えます。

1. メイン画面の［ファイル］メニューから［計測対象一覧ファイルを開く］コマンドを実行します。
2. ［ファイルを開くダイアログボックス］で読み込むファイルを指定します。
3. ［ファイルを開くダイアログボックス］の［開く］ボタンにより読み込まれます。
4. メイン画面の［計測］メニューから［計測］コマンドを実行します。

10.1.6 計測結果の判定について

計測結果の判定については、［測定値の判定］を参考にして下さい。

また、アラーム指定の基準値の設定については［アラーム指定基準値］を参考にして下さい。